Peer to peer networking in Ethernet broadband access networks

AYODELE DAMOLA

KTH Information and
Communication Technology

Master of Science Thesis
Stockholm, Sweden 2005

IMIT/LCN 2005-10

Peer to peer networking in Ethernet broadband access networks

by
Ayodele Damola
Ericsson AB and KTH
ayodele@kth.se

27 May 2005
Stockholm, Sweden
A thesis presented to the Royal Institute of Technology, Stockholm
in partial fulfillment of the requirement for the degree of
Master of Science in Internetworking

Academic Advisor and Examiner:
G. Q. Maguire Jr.
School of Information Technology and Communication (ICT)
Royal Institute of Technology (KTH)

Industry Supervisor:
Hans Mickelson
Ericsson Research
Broadband Access – Networks & Technologies
Ericsson AB

Signature:
_____
Date:
_____

Signature:
_____
Date:
_____

Abstract

The use of peer-to-peer (P2P) applications is growing dramatically, particularly for sharing content such as video, audio, and software. The traffic generated by these applications represents a large proportion of Internet traffic. For the broadband access network providers P2P traffic presents several problems.

This thesis identifies the performance and business issues that P2P traffic has on broadband access networks employing the McCircuit separation technique. A mechanism for managing P2P within the access network is proposed. The P2P diversion algorithm aims to manage P2P traffic within the access network based on layer 2 and layer 3 information without employing intrusive layer 7 traffic detection. To solve the contention problem experienced by best effort traffic in the access network, a solution based on the diversion algorithm and on a QoS based traffic classification scheme is proposed. A business model defining the business roles and pricing schemes is presented based on the features offered by the P2P diversion algorithm introducing new opportunities for gaining revenue from P2P traffic for the network service providers and providing better services to users.

Abstract in Swedish

Användningen av peer-to-peer (P2P) applikationer ökar dramatiskt, speciellt för spridningen av video, musik, och mjukvara. Trafiken som skapas av dessa program utgör en stor del of trafiken på Internet. För bredbandsaccess operatörer ställer P2P trafik många problem.

I detta examensarbete så identifieras både de egenskaper och affärsaspekter som P2P trafiken har på ett bredbandsaccessnät som använder McCircuit som seprationsmekanism för trafiken mellan användare och en mekanism, "peer-to-peer diversion mechansim" (P2PDA), för att hantera P2P trafiken i ett McCircuit baserat accessnät beskrivs. P2PDA algoritmen hanterar P2P trafik i accessnätet baserat på lager 2 och lager 3 information utan att ta hänsyn till applikationslagret (Lager 7). För att få en bra fördelning mellan best-effort trafik och prioriterad trafik så föreslås en lösning baserad på kombinationen av P2PDA och QoS baserad trafik klassificering. Slutligen så defineras en affärsmodell där affärsroller och olika varianter på prissättning för P2P diskuteras baserad på de egenskaper som den förslagna algoritmen medför och den ekonomiska vinst som denna lösning medger.

Acknowledgements

Thanks to my industrial supervisor at Ericsson Mr. Hans Mickelsson first for selecting me for this thesis project and for his continuous help and support along the whole way. Thank you Hans. Thanks to Mr. Jan Söderström for the opportunity to do my project at Ericsson Research. Thanks to Mr. Torbjörn Cagenius for your technical advice and deep insights freely given during our many discussions. My knowledge of broadband networks has been broadened by his input. Thanks to Mr. Zere Ghebretensaé for taking time to discuss several aspects of the MUSE project and for supporting my work in general. A big thanks to Mr. Jonathan Olsson who's help make possible the practical implementation of my ideas. His input gave me an understanding of the functionality of hardware and software used in my project. I would also like to thank Mr. Panagiotis Saltsidis for providing me with the statistical data that I used to further establish my claim of the significance of P2P traffic in broadband access networks. Thanks to Mr Johan Kölhi for validating my ideas and providing insight into some technical implementation issues.

Finally, I would like to extend my appreciation to my supervisor at KTH, Professor Gerald Q. Maguire Jr. His continuous feedback on my progress shaped my project making it a worthy academic work. Thank you very much professor for all your advices and comments.

Table of contents

LIST OF FIGURES

LIST OF TABLES

Acronyms

ADSL	Asymmetric Digital Subscriber Line
AN	Access Node (Penult)
ANP	Access Network Provider
AS	Autonomous System
ASP	Application Service Provider
BRAS	Broadband Remote Access Server
CAIDA	Cooperative Association for Internet Data Analysis
CPE	Customer Premises Equipment
CPN	Customer Premises Network
DRG	Digital Residential Gateway
DSLAM	Digital Subscriber Line Access Multiplexer
DWDM	Dense Wavelength Division Multiplexing
ELN	Ethernet Local Node
EN	Edge Node (Apex)
IP	Internet Protocol
ISP	Internet Service Provider
MAC	Media Access Control
MACFF	MAC forced forwarding
McC	McCircuit
MTU	Maximum Transfer Unit
NP	Network Processor
NSP	Network Service Provider
OSGi	Open Services Gateway Initiative
P2P	Peer to peer
P2PDA	P2P diversion algorithm
PAMP	Penult-Apex Messaging Protocol
PMP	Paris Metro Pricing
QoS	Quality of Service
RIAA	Recording Industry Association of America
RNP	Regional Network Provider
SA	Service agent
SLA	Service Level Agreements
SP	Service Provider
WFQ	Weighted Fair Queuing
WRR	Weighted Round Ribbon
VLAN	Virtual Local-Area Network
VoIP	Voice Over IP

CHAPTER 1. INTRODUCTION TO PEER TO PEER

1.1 Definition of peer to peer

The Peer-to-peer (P2P) communication paradigm is a distributed computing approach where each node or peer acts as both a client and a server of a resource. In a P2P file-sharing application, for example, a peer both requests files from its peers, and stores and serves files to its peers. The following figure shows the basic node message exchanges in a pure P2P system.

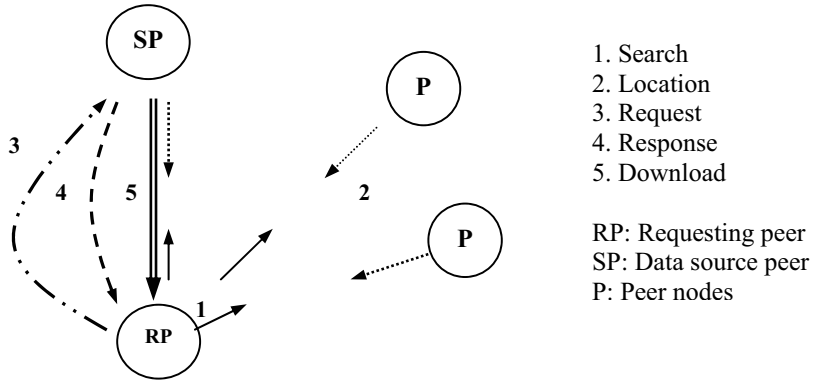


Figure 1. Interaction of P2P nodes

More formally, "the term 'peer-to-peer' refers to a class of systems and applications that employ distributed resources to perform a function in a decentralized manner" [3]. The resources encompass computing power, data (storage and content), network bandwidth, and presence (computers, human, and other resources). The function can be distributed computing, data/content sharing, communication and collaboration, or platform services. Decentralization may apply to algorithms, data, and meta-data, or to all of them. This does not preclude retaining centralization in some parts of the systems and applications if it meets the requirements of these systems or applications. Typical P2P systems reside on the edge of the Internet or in ad hoc networks.

"P2P is a class of applications that takes advantage of resources – storage, cycles, content, human presence – available at the edges of the Internet. Because accessing these decentralized resources means operating in an environment of unstable connectivity and unpredictable IP addresses, P2P nodes must operate outside the DNS system and have significant or total autonomy from central servers" [10].

1.2 Taxonomy of P2P computing

Several classification schemes of the P2P paradigm are presented below. The classification schemes each view the P2P paradigm from a different perspective. The first classification views P2P computing based on the degree of decentralization compared to the traditional client-server architecture. The second scheme classifies P2P based on network structure. The last scheme presents the different classes of P2P applications.

1.2.1 Taxonomy based on degree of centralization

According to the classification made in [2], P2P architecture, file sharing architectures in particular, can be classified by their 'degree of centralization', *i.e.* to what extend they use the client/server model to facilitate the cooperation between nodes.

1.2.1.1 Pseudo-centralized

In this architecture, there is a server (or a cluster of servers) that facilitates the cooperation between peers and can even provide a service such as file lookup. The pseudo-centralized architecture utilizes a client-server network structure. This was the architecture used by the Napster system [22] and has proved to be less resilient to failures than the two other approaches (the Napster service was closed by shutting down the Napster servers). However, some modern P2P systems use a similar approach with the modification that servers are numerous, geographically distributed, and interconnected. This is for example the case of the eDonkey system [37]. In systems like Napster and Seti@Home [21] coordination between peers is controlled and mediated by a central server, although the peers may also contact each other directly. This makes these systems vulnerable to the problems of centralized servers. To overcome the limitations of a centralized coordinator, different hybrid P2P architectures [4] have been proposed to distribute the functionality of the coordinator over multiple indexing servers that cooperate with each other to satisfy user requests. DNS is another example of a hierarchical P2P system that improves performance by defining a tree of coordinators, with each coordinator responsible for a peer group. Communication between peers in different groups is achieved through a higher-level coordinator.

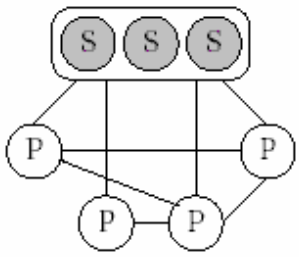


Figure 2. Pseudo-centralized P2P architecture

1.2.1.2 Purely decentralized

In these architectures, all nodes have the same responsibilities, regardless of their capacities, location, or provided resources. All nodes perform the same tasks, acting both as server and client, without any central coordination. Hosts participating in such networks are called *servents* (SERVer and cliENT). This architecture was used in the original Gnutella [23]. It is no longer heavily used because it is generally quite inefficient due to its approach of flooding requests when searching for content. Messages may have to cross a large number of hosts before reach an adequate peer (a peer possessing a given file for example). This increases response time. It is also difficult to provide guarantees in purely decentralized networks, for example, it is difficult to ensure the network is not fragmented. Fragmentations occurs when nodes with inadequate bandwidth become chokepoints that partition the P2P network into several disconnected components.

Purely decentralized systems (e.g. Gnutella and Freenet [38]) use message forwarding mechanisms to search for information and data. The problem with this is that they end up sending a large number of messages over many hops from one peer to another. Each hop contributes to an increase in the bandwidth used on the communication links and to the time required to get results for the queries. The bandwidth used for a search query is proportional to the number of messages sent, which in turn is proportional to the number of peers that must process the request before finding the data. Due to the flooding (broadcast) of requests in purely decentralized systems the numbers of messages generated is immense. Once the peer with relevant content is found a HTTP connection is established and the HTTP GET command is used to download the file, then the amount of traffic generated is relative to the size of the file.

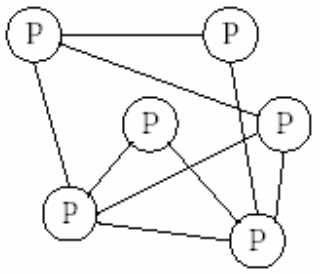


Figure 3. Purely decentralized P2P architecture

1.2.1.3 Partially centralized

In these systems, some nodes assume more responsibilities than others, acting as local servers for files shared by local peers and providing connectivity with other '*supernodes'*. The resulting P2P architecture forms a (two-level) hierarchy with better performance and scalability than the purely decentralized model. It is used in modern file sharing systems such as FastTrack [41], Kazaa [40], iMesh [39], or the new version of Gnutella.

Depending on the P2P system, supernodes could be elected dynamically (in some systems the user has an option to switch off supernode mode) thus avoiding a single point of failure (they are replaced if they become unavailable).

In Gnuttella, for example, when a host with enough CPU power joins the network, it automatically becomes a supernode (also called superpeer) and connects to other superpeers forming a flat unstructured overlay network of superpeers. If it receives connections from a sufficient number of client nodes, then it remains a superpeer; otherwise, it turns into a regular client node. If it later cannot connect to any superpeer (e.g. all have reach maximum client capacity), it again tries to become a superpeer for another probation period.

In the file sharing P2P application Kazaa, any computer can become a supernode if it has sufficient computing resource and a broadband connection. Being a supernode does not affect performance noticeably because the computing effort is limited to 10% of the CPU power available, but it can stress the upstream link for users of asymmetric links (such as ADSL).

A supernode indexes the content of client nodes. This is done when other users in the neighborhood upload to the supernode a list of files they are sharing, whenever possible

using the same ISP or located in the same region as the supernode. This feature is implemented in DC++ [51] which enables users to choose supernodes called hubs that are in their ISP's network. When one of these users searches for a file, a search request is sent to the supernode. The supernode then searches its list of files to find neighbors possessing files matching the query. The search request could be forwarded to other supernodes if there is no matches found locally. The results are then sent back to the client node which made the query. The actual download will be directly from the computer that has the file, rather than from the supernode.

A client node keeps only a small number of connections open and each of these connections is to a supernode. This has the effect of enabling network scaling, by reducing the number of nodes involved in message handling and routing, as well as by reducing the actual volume of traffic among them. Because of these super-nodes, which also act as search hubs, the speed with which queries are answered within the controlled framework is comparable to a centralized network model.

The difference between the partially centralized and pseudo-centralized architecture resides in the software. In partially centralized systems, supernodes are elected dynamically and are also peers (e.g., they also download files), cooperation between peers is a 'part-time activity'. In pseudo-centralized networks, the client and server software are generally different. Performance may be better, as compared to decentralized systems, because servers are generally dedicated to cooperation between peers only (it is a 'full-time activity'). On the other hand, the system is less flexible and fault-resilient than in the partially centralized case.

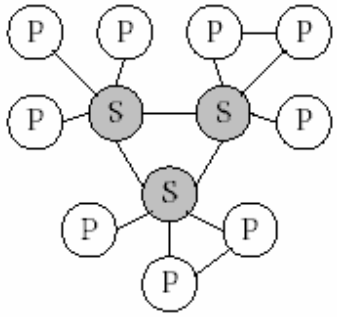


Figure 4. Partially centralized P2P architecture

1.2.2 Taxonomy based on Network structure

Soldani [2] classifies P2P systems into three groups, regarding their level of structure.

1.2.2.1 Unstructured networks

In unstructured networks, the placement of data is completely unrelated to the overlay topology. Overlay networks are virtual communications structures that are logically 'laid over' a physical network such as the Internet. They provide application-level functionality that is out-of-scope for the underlying network. Since there is no way of knowing where a given resource is *a priori*, searches are conducted at random, asking a number of servents if they have files matching the query. These servents may ask their own neighbors about the resources eventually giving the originator a way to access the entire P2P system (possibly by asking every node taking part in the system). Although there are different

possibilities for the construction of the overlay network and for the query mechanism, unstructured networks generally result in poor lookup performance, scalability problems, and inefficient network usage.

However, this scheme is the most widely used since it easily accommodates a transient population and is well adapted to file-sharing. Users of such systems want some specific file(s) and don't want to store other files for the sake of system efficiency; they don't want to be concerned with issues such as lookup performance (even if they prefer it to be fast); or redundancy (even if they want high availability). To solve performance and scalability issues in unstructured networks, a partially centralized model can be used. Searches are still conduced at random, but only at the supernode/server level. End users only send queries to their local supernode/server. This two-level structure improves performance and scalability, making these unstructured networks viable. The price is that the uplink of the supernode could become the bottleneck of the system as all signaling is done via it.

1.2.2.2 Structured

Structured networks have mainly emerged in the academic world. In such systems, topology is closely related to hosts' content or host content is related to topology. Files (or pointers to files) are stored at specific locations in the P2P system and a mechanism is provided to map a file identifier to its location (or the location of its pointer). Using a distributed routing table (which generally uses hash tables), queries can be forwarded to a suitable host much more efficiently than in the unstructured case. The disadvantages of structured networks are the difficulty of maintaining the routing table with frequent arrivals and departures of peers and mapping a keyword query to a unique file identifier. The frequent arrival and departure of hosts is related to random user behavior of connecting to the P2P system. Structured networks such as Chord [18], CAN [17], or Tapestry [19] will not be extensively discussed in the remainder of this text since they have little impact on network traffic and P2P behavior detection (due to their small user bases).

1.2.2.3 Loosely structured

Loosely structured networks are hybrid solutions between structured and unstructured networks. In such systems, a mapping exists between file location and topology, but it is not completely specified and may result in search failure (the search is then conducted as if the network was unstructured). Freenet [20] is an example of such a loosely structured network.

1.2.3 Taxonomy of P2P applications

According to [3], three main classes of P2P applications have emerged: parallelizable, content and file management, and collaborative. Figure 4 shows the kind of applications that fall into each of the classes.

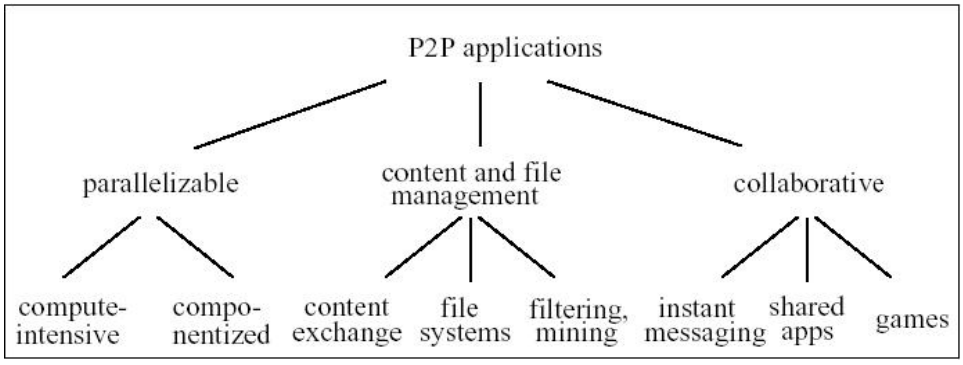


Figure 5. Taxonomy of P2P applications [3]

1.2.3.1 Parallelizable.

Parallelizable P2P applications split a large task into smaller sub-pieces that can execute in parallel using a number of independent peer nodes. Most implementations of this model have focused on compute-intensive applications. The general idea behind these applications is that idle cycles from any computer connected to the Internet can be leverage to solve difficult problems that require extreme amounts of computation. Most often, the same task is performed on each peer using different sets of parameters.

Examples of implementations include searching for extraterrestrial life [21], code breaking, portfolio pricing, risk hedge calculation, market and credit evaluation, and demographic analysis. Componentized applications have not yet been widely recognized as P2P. However, [3] envisions applications that can be built out of finer-grain components that execute over many nodes in parallel. In contrast to compute-intensive applications that run the same task on many peers, componentized applications run different components on each peer. Examples include Workflow, JavaBeans, or Web services in general.

1.2.3.2 Content and file management.

Content and file management P2P applications focus on storing information on and retrieving information from various peers in the network. The model that popularized this class of application is the content exchange model. Applications such as Napster [22] and Gnutella [23] allow peers to search for and download files, initially primarily music files, that other peers have made available. For the most part, current implementations have not focused on providing reliability and rely on the user to make intelligent choices about the location from which to fetch files and to retry when downloads fail. They focus on using otherwise unused storage space as a distributed content/file server for other users. These applications could ensure reliability by using more traditional database techniques such as replication. A number of research projects have explored the foundations of P2P file systems [17, 18]. Finally, filtering and mining applications such as OpenCOLA [42] and JXTA Search [30] are beginning to emerge. Instead of focusing on sharing information, these applications focus on collaborative filtering techniques that build searchable indices over a peer network. A technology such as JXTA Search can be used in conjunction with an application such as Gnutella to allow more up-to-date searches over a large, distributed body of information.

1.2.3.3 Collaborative

Collaborative P2P applications allow users to collaborate, in real time, without relying on a central server to collect and relay information. Instant messaging is one subclass of this class of application. Skype [32] is an example of such a service. Similarly, shared applications that allow people (e.g., business colleagues) to interact while viewing and editing the same information simultaneously, yet the users are possibly thousands of miles apart, are also emerging. Examples include Buzzpad [31] and distributed Power-Point [43]. Games are a another type of collaborative P2P application. P2P games are hosted on all peer computers and updates are distributed to all peers without requiring a central server. Example games include NetZ 1.0 by Quazal [33], Scour Exchange by CenterSpan [44], Descent [34], and Cybiko [35].

1.3. Traffic characteristics of P2P applications

1.3.1 High bandwidth usage

Peer-to-Peer-traffic has become a major part, sometimes even the dominant part of current networks. The impacts of Peer-to-Peer traffic can be clearly observed. In 2002-2003, 70% of the overall traffic in the German research network was already due to Peer-to-Peer applications while, in the Abilene backbone 30% to 60% of the overall traffic is caused by Peer-to-Peer applications [5].

According to the Cooperative Association for Internet Data Analysis (CAIDA) [12], service provider network traffic is dominated by peer-to-peer file sharing applications. P2P applications generate two types of network traffic: overhead traffic (searches and keep-alives) and data traffic (file transfers).

For April 2003, according to Sprint's IP Monitoring Project [26], for the majority of the monitored links in New York and San Jose, P2P traffic is approximately 20% of the total volume. In February 2004, 25-40% of total bytes corresponds to P2P traffic. The variance is due to port hopping behavior of P2P applications and measurements made using Coral Reef [59] application port tables. This data can be interpreted as P2P activity increasing in 2003-2004.

1.3.2 High signaling load

An experiment detailed in [9], showed that while HTTP traffic is asymmetric in nature, P2P traffic is symmetric. This is attributed to almost similar rates of both upstream and downstream flows. A detailed analyses of the P2P traffic showed that while a portion of it was as a result of file transfer (which is naturally expected), a large amount of P2P traffic is signaling overhead.

The reason for this large signaling overhead is the change of the semantic role of the Internet. The requested content and its large number of replicas are distributed over a tremendous numbers of nodes at the edge of the network. For reliability reasons most P2P networks avoid use of central lookup servers, unlike Napster. Thus a distributed search on a number of nodes is necessary to find a replica of the desired data [9]. Hence, query and node keep-alive messages constitute a big portion of P2P traffic.

1.3.3 P2P locality

P2P traffic has increased the amount of traffic between users in a significant way. When two or more P2P clients start using the network they form a direct connection to exchange files. Whether the clients use the same or different providers is not a determining factor in how the P2P connections are made. P2P file exchange has significantly increased the potential for in true autonomous system (AS) traffic. As observed in [6]: Peer-to-peer traffic does not show strong signs of geographic locality because the peer-to-peer applications do not exploit topological locality. In Gnutella, each peer has a user-driven neighbor table to locate a file. A file request is spread out through neighbors and each peer receiving the request checks its local published files. So the requested file is downloaded without respect to physical network proximity but rather based on only the AS network topology. In the figure 6a, the overlay presents node N with a list of peers: peer 1 and peer 2, which have the desired content. If the content is downloaded from peer 2 as in figure 6b, unnecessary inter-AS traffic is generated as opposed to getting the content locally from peer 1.

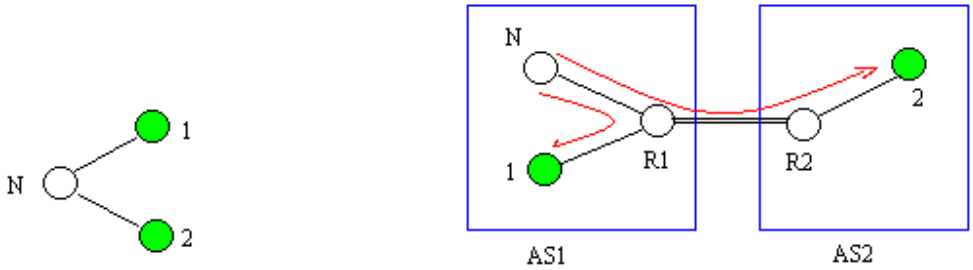


Figure 6. a) Overlay model view of peers b) Underlying physical topology

Additionally Sen and Wang observe that 80% of the ASs communicate with multiple ASs, and the top 1% of the ASs communicate with at least 476 other ASs [7]. This inter-AS traffic is especially significant to ISPs, as it typically affects their bottom line. The conclusion is that, although there is some evidence for weak locality at a large spatial scale, P2P applications do not **yet** exploit such information on a large scale, and consequently, P2P traffic does not show strong signs of geographic locality. Developments such as the KazuperNode tool [8] provide methods for selecting the super-node to which one connects. On the one hand this could potentially increase locality if users tended to connect to topologically nearby super-nodes. On the other hand, there could be less locality if users connect to non-local super-nodes in their attempts to locate content. However, the tool does provide locality information based on IP address, city, state and country.

There are some researches that have proposed adding additional overlays to reduce the physical routing delay. Brocade [45] uses a landmark routing overlay in which selected high capability peers near the network access points provide a shortcut route across distant network domains. Expressway [46] also organizes a secondary overlay on the basis of actual network topology. These secondary overlays reduce the routing delay occurred in a logical hop to some extent.

1.3.4 Upstream / Downstream Traffic Ratio disproportion

Broadband access networks are often asymmetric in nature: the amount of traffic that a network can sustain upstream, is different from the amount it can sustain in the opposite direction. This may cause traffic congestion and unutilized capacity. P2P applications encourage users to share files, thus a typical peer serves gigabytes of files. This may cause a drastic change in the upstream/downstream ratio, and as a result congestion on the upstream link leads to high packet loss.

1.3.5 Zipf-like popularity trends of P2P objects

It has been observed in [53,55] that many document storage systems, including the WWW, exhibit Zipf-like distributions on the popularity of documents. This reflects the fact that some popular documents are very widely copied and held, while most documents are held by fewer peers. The same can be said of content categories: there are some content categories (such as "Top 40 Hits" in the music domain") which are very popular and widely held, while most other categories (such as "Acid Jazz") are less widely held [54]. From this it can be inferred that in a P2P system files with different popularities exist within each content category, governed by a Zipf-like distribution.
A study at Carnegie Mellon University [65] made traces of Gnutella queries and the results a shown in figure 7.

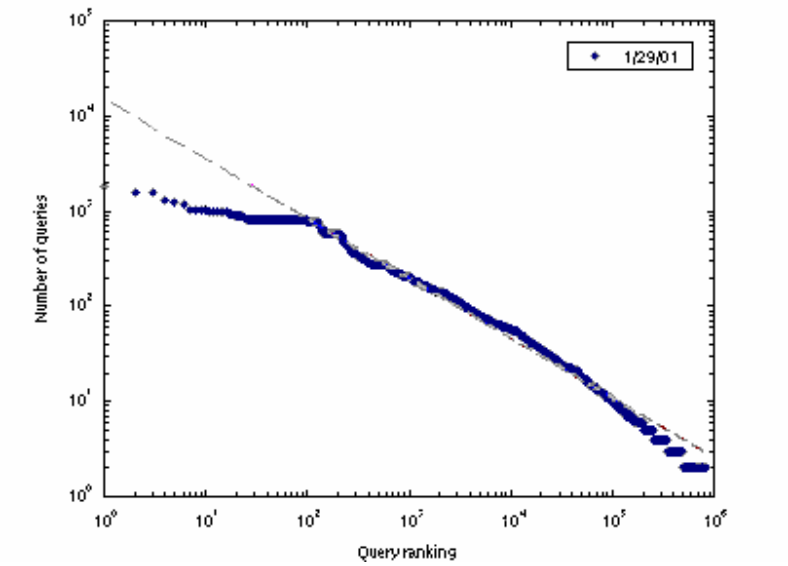


Figure 7. Frequency of query string observed versus query ranking [65]

The figure shows the number of times a query is observed versus the ranking of the query on a logarithmic scale. Rank 1 is the most popular query. If each curve were to be a straight line, then the popularity of queries follows a Zipf-like distribution with the probability of seeing a query for the **i'th** most popular query is proportional to **1/(i^alpha)**. The curve looks like two straight lines with an inflection point at around query rank 100. The first portion of the curve for queries rank 1 to 100 is flatter. This implies that the most popular queries are almost equally popular. The second portion of the curve, after query rank 100, fits a straight line reasonably well. The conclusion is that

very popular documents are equally popular, while less popular documents have a distribution which follows a Zipf-like distribution.

1.4. Trends and statistics of P2P applications

Current trends suggest that P2P applications are used mainly for file sharing. Figure 8 gives the worldwide distribution of active P2P users categorized by the P2P applications.

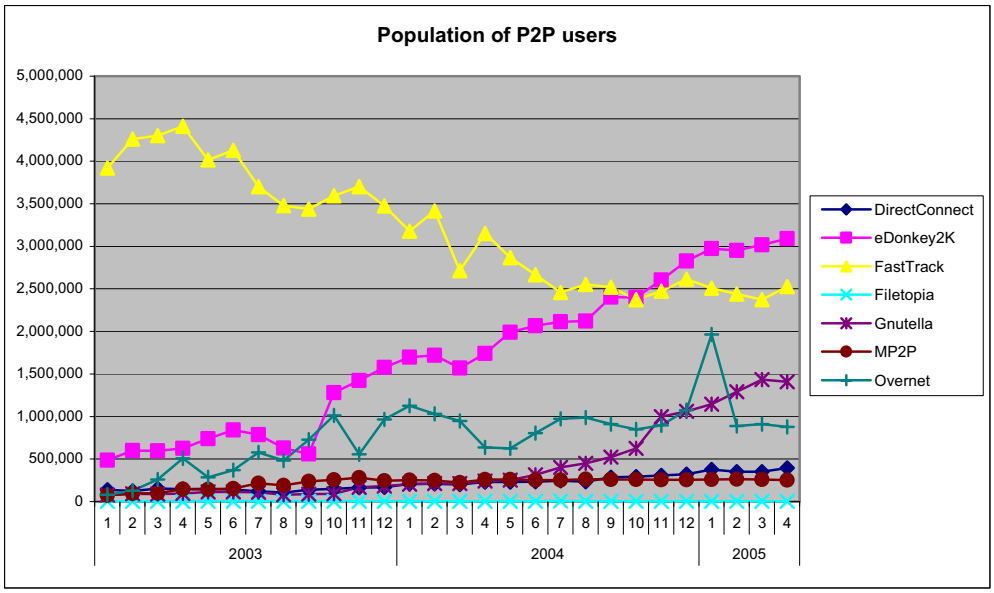


Figure 8. Worldwide population of active P2P users [56]

The ongoing battle between users of file-sharing programs and media copyright-enforcement organizations (most notably the Recording Industry Association of America (RIAA)) has seemingly become a ping-pong match of lawsuits, threats of lawsuits, countersuits, office raids of commercial P2P services, and soda pop promotional gimmicks encouraging people to download music from legal music downloading services.

Regardless of all the threats, intimidation, and spoofed music files clogging networks, P2P services in which users engage in file sharing continue to thrive. Activity on them still far surpasses the traffic of the legal music download sites, such as iTunes Music Store and the now legit Napster. One weakness of some P2P networks is the fact that it's so easy to identify a user's IP address. The RIAA has managed to use such extracted information to subpoena ISPs for the identities of potential defendants.

The current trend in the P2P community is to use applications that allow file-sharing without revealing the identity of the users to each other or to the rest of the network, here anonymous identity is achieved by encryption and hiding users' IP addresses. Examples of such applications are FreeNet, Mute, Ant, and Winny. The term ‘Freenet’ has emerged which promotes anonymous and encrypted P2P file sharing applications. Apparently this is in an effort to evade lawsuits due to copyright infringement. The distribution by the

country of the global population of Freenet users as of late May 2004, is presented in figure 9.

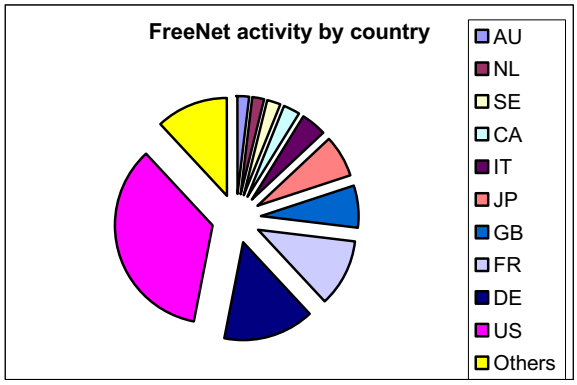


Figure 9. Freenet activity by country [28]

The trend of P2P users migrating to anonymous P2P is expected to increase as more P2P applications implement encryption.

Apart from file-sharing applications, a gaining niche for P2P is VoIP. Skype [32] is a free instant-messaging P2P software that supports VoIP. Today Skype has approximately 30 million registered users [57], it has served about 2.7 billion call minutes and has multiple OS support. Skype appears to have penetrated 20% of its potential market, and with around 2 million concurrent users, more than 1% of the world's broadband population is running Skype at any given time. With introduction of video-conferencing service (For example, broadband provider Bredbandbolaget [50] started offering all mass TV channels in mid 2005), the bandwidth consumption of the application is expected to increase.

1.5. Impact of P2P traffic on broadband service providers

The impact of P2P traffic on the Internet in general and on broadband access networks in particular is significant. The following subsection gives a summary of this.

1.5.1 Bandwidth issues

Broadband access is widely based on xDSL technology. The most widely deployed type of DSL to residential customers is Asymmetrical DSL (ADSL). As the name implies, the downstream capacity is higher then upstream reflecting higher consumption of content than generation. The most common delivery model is based on offering internet access, by setting up one Permanent Virtual Channel (PVC) for each user. This channel then functions as a “best-effort” transport medium for all devices and services in the home. This delivery model is data centric and was designed with generous levels of oversubscription, leading to possible packet-loss from network congestion. It offers no prioritization of content and utilizes the statistical nature of traffic to multiplex and aggregate traffic from many users onto a common second mile link that has much lower capacity than the sum of dedicated first-mile capacity.

P2P has the most impact of any type of traffic today on the bandwidth of broadband networks. If the typical P2P file is one thousand times the bandwidth of regular World Wide Web traffic, and this traffic becomes the primary traffic on the Internet, then P2P traffic will significantly impact Service Providers networks which rely on

oversubscription, subtended digital subscriber line access multiplexers (DSLAMs), and inverse multiplexing over ATM (IMA) trunks.

The challenge faced by service providers is to build or evolve an existing network so that it is dynamic enough to grow as the traffic demand grows. This growth would require reducing the oversubscription ratio and having enough bandwidth available to be used as needed. There is significant cost associated with building excess bandwidth [11].

Most DSL and cable providers built data networks and billing models around asymmetrical services. P2P networks changed this model. Now average desktops are not just clients on the Internet but are functioning as servers and file depositories. In the past, customers had more download capacity than upload. Now customers require more symmetric data models to support high upload and download speeds.

Traffic of P2P applications is classed as best effort traffic. File-sharing P2P application create contention in the best effort traffic class. Non-P2P best effort traffic suffers from undesirable delay and packet loss due to QoS bandwidth policing during peak hours.

1.5.2 Additional Internet transit fees

Bandwidth usually isn't free. With P2P applications sending a high volume of bits in both directions, there is likely more transit fees being paid than truly required. Due to weak locality of P2P traffic, much of the traffic that could be internal is going external. Depending upon the situation of the broadband network operator, it may be better to encourage subscribers to download their content from another local subscriber rather than fetching it from some other peer [11].
Costs are a primary concern to service providers. Below are a few of the many costs associated with unrestrained P2P traffic [14].

- Costly bandwidth consumed – on a typical service provider network, over 60% of total bandwidth is used by P2P traffic [14]. This traffic is comprised of "protocol-chatter" as well as the transmission of the shared files themselves.
- Additional network transit costs occur, as P2P traffic connects in an ad hoc fashion hence subscribers are as likely to download a file from halfway around the world as they are to download it from their neighbor.
- Over-subscription business model undermined – a common business model among service providers; over-subscription is unworkable when 10%-20% of the users consume 80% of bandwidth and this type of users are increasingly common.
- Loss of brand equity – in today's competitive broadband industry, a congested service provider network translates into churn as subscribers change Internet providers.

In summary P2P network traffic consumes a large portion of bandwidth, and as P2P application usage continues to increase, so do service providers' Internet transit charges. P2P growth also affects Quality of Service (QoS) for all subscribers and often causes unplanned network expenditures [13].

1.5.3 Evolution of billing models

The upstream/downstream traffic ratio equality, as mentioned in section 3.4, could affect the billing models currently used by Service Providers [27]. In the past, customers'

network usage was predictable; therefore Service Providers were able to create effective billing models for various data rates and services. P2P creates virtual supercomputers and file systems with no geographic boundaries or central administration. P2P has no common domain to bill for usage, therefore new billing models will have to be created to recoup cost of supporting this type of network. The current model of offering small upload speeds and greater download speeds in DSL broadband may no longer be valid. With P2P, individual desktop computers are functioning as servers and clients and therefore require more symmetrical data rates. The current model of selling symmetric and high bandwidth services only to businesses may have to be reviewed as P2P grows.

1.5.4 Security issues

In addition to bandwidth issues caused by P2P traffic, Service Providers also have to face security issues [27]. According to an article published by Sandvine in [19], research shows that file sharing networks will become the most efficient means of spreading worms and will have the largest potential of exhausting service providers' network. Therefore, the Service Provider will have to implement more stringent virus detection and isolation methods as well as access controls mechanisms.

1.6 Methods of Control

Today the P2P overlay network has no relation to the underlying physical topology of the network which leads to large inefficiencies where content is being sourced and causes additional packet traversals over links. Several papers [1, 2, 6] and Internet traffic logs [5] suggest that the bandwidth intensive nature of P2P applications has significant impact on the underlying network. Below are some methods of P2P traffic control.

1.6.1 Traffic blocking

P2P blocking refers to the practice of blocking ports at the network access point (e.g. DSLAM) that are commonly used by the most popular P2P networks. The aim of P2P blocking is to reduce bandwidth usage by blocking all P2P traffic, and in so doing, completely avoid the typical costs of P2P usage [14], but at a direct cost to the users.

However, P2P applications have rapidly evolved such that accurately accounting for their traffic is more difficult. In particular, previously the applications used default static TCP ports, and it was possible to account for the bulk of the P2P traffic by monitoring a relatively small number of ports. The current trend is that well know and registered ports are not defined or used by all applications, this especially true of P2P applications. Furthermore, in some cases server ports are dynamically allocated as needed (for instance, one might have a control connection on which a data port is negotiated, as FTP does). Finally, the use of firewalls to block unauthorized and unknown applications from using a network has spawned work arounds that have made the mapping from port number to application ambiguous. Such port-hopping makes any limitations based on mapping exceedingly impractical [6].

The alternative is to track a larger number of ports that contribute significant traffic volumes and that are *suspected* to carry P2P traffic. The problem with this approach is that (i) it may not be feasible to track such a large and potentially dynamic set of ports, and (ii) such widespread rate control may adversely affect the performance of many non-

P2P applications on these other ports – this would be undesirable for the customers of the broadband providers.

1.6.2 Traffic shaping

Shaping refers to the practice of processing, buffering, and prioritizing all traffic traveling through the network access point. This potentially allows a service provider to give priority to non-P2P traffic, leaving whatever bandwidth is left over for P2P. Each individual data packet that arrives at the access node is examined and classified based on an identification key found in the packet. Based on the priority of each category of traffic, the packets are then entered into a queue and transmitted. In a P2P-shaping context, P2P packets are sent last, consuming whatever bandwidth is left over after all the higher priority traffic has been sent [14].

Shaping certainly has its advantages; a service provider can gain a degree of control over their network, by prioritizing their traffic to suit their subscriber base and cost concerns is a useful tool. Associated P2P costs can be reduced in a way that avoids the sizeable pitfalls of completely blocking P2P traffic.

However, because shaping relies on accurately identifying packets as P2P, it is susceptible to a range of evasion tactics implemented by P2P developers. The most widely used approach is encryption, which hides all details of the P2P protocol, making it impossible to detect.

The limitation of traffic shaping is that it can only provide temporary relief since it doesn't do anything to help improve the overall efficiency of the P2P overlay network's use of sources, other than limit the amount of traffic. This limitation is translated into possible subscriber dissatisfaction through slower file downloads.

1.6.3 Rate limiting

Rate limiting is implemented by controlling the rate at which data can flow into or out-of the network. The effect of these limits is to shape the instantaneous traffic peaks. Despite this, caps have been widely used by the industry and seem partially successful. However, P2P traffic is a relatively "passive" traffic source, as the requester can queue-up a set of requests for files, then walk away. The file provider does not even need to be at their PC, their application can serve requests in the background. In this situation rate capping will simply make the requests take longer, but is unlikely to change the behavior of P2P participants [6].

1.6.4 Over-provisioning and topology upgrade

When a network is regularly overwhelmed with traffic, a common approach is to obtain more bandwidth by purchasing it from a larger provider and upgrading the existing infrastructure to handle the increase. To a certain extent, this is logical: if the present amount of bandwidth is not enough to handle traffic volumes, then additional bandwidth is required. If the service provider is in a growth phase, then a solution that facilitates that growth is appropriate [14].

However, while acquiring more bandwidth and building up infrastructure does provide more bandwidth, it does nothing to mitigate the problems associated with P2P. In fact, the increased amount of bandwidth actually encourages increased P2P traffic, as the

subscribers have increased resources to consume; the more that is provided, the more is consumed, while the associated costs of P2P only increase [14].

Node splitting, higher capacity links and faster routers all help in provisioning higher average bandwidth to the subscribers and could improve the end user experience. Lowering the number of subscribers per uplink via node splitting is practical for some operators to decrease the level of over subscription. Node splitting is mostly used in optical fiber networks with the use of DWDM [47] (Dense Wavelength Division Multiplexing) channel upgrade. DWDM increases bandwidth in legacy systems by combining and transmitting multiple signals simultaneously at different wavelengths on the same fiber.

The limitation of upgrading the network is that it will not help manage costs and the problem of inefficiencies in the P2P overlay network will still remain, although the magnitude of this problem will be lower as now the generated traffic doesn't cause the same strain on the links.

1.6.5 Tiered services

As the broadband industry matures, one-size-fits-all products lose their ability to sustain demand. Demand exists for both premium and value tiers of broadband, defined primarily by speed. The ability to support several unique service levels becomes important when combining disparate end users who range from full-fledged businesses, home office users, and simple home users who surf the Internet. Tiered service is supporting several classes of service, each with unique service level demands and characteristics. By providing Quality of Service (QoS) metrics into the DSL access element, service providers can assign unique classes and QoS levels to individual customers. Rate limiting and policing of the established traffic parameters are critical in a tiered service.

Tiered services are implemented be having service differentiation. Service differentiation is based on setting up separate virtual channels (VLANs) with a QoS setting for each service and assigning services to specific ports on the Customer Premises Equipment (CPE). Service differentiation is then transparent to the devices and performed on port level or at packet level. Both the CPE and DSL access multiplexer (DSLAM) then prioritize delay sensitive traffic such as voice and video before data. The tiered solution can support multicasting and therefore allows services such as VoD, IPTV and VoIP. This scenario is the preferred situation of many ANPs as it grants them sole access to differentiated services [63]. P-Cube offers a solution [64] that goes from selling just connectivity and bandwidth to selling services, and application performance on a tiered basis.

A P2P service in a tiered service architecture was outlined by a research group at BT which proposed a solution to build a network that encourages legal peer-to-peer trading where money goes to the appropriate content owner, while at the same time making illegal video trading sufficiently slow or expensive so as to discourage it [1]. The approach entails creating an underlying network topology aware P2P application, which will offer content download at different rates depending on price.

This approach has two major challenges: The first is to encourage all users to use this P2P application rather than standard P2P applications. The other is their business model, which allows the network operator and the content owner to share revenue.

1.6.6 Caching

Depending upon the requirements, establishing a large cache of popular content in the network may be effective if the network can also be trained to utilize the cache server [13]. The search for content is done first in the cache, thus reducing downstream traffic. If the cache doesn't have the required data, then the search is performed the usual way. The cache is automatically seeded; when a user requests a file and that cache does not have, it makes the connection to the source of the content and retrieves the file, simultaneously storing it on its local drive and sending it to the requesting user [14].
However, the foremost concern for service providers is the legality of such a solution; the access network provider would no longer be merely providing basic connectivity, but potentially providing copyrighted content as well. Caching content brings up a number of copyright issues that most likely will prevent any operator from implementing this alternative. Legal concerns would not be a problem if the content is encrypted and the cache operator does not have the key.

1.6.7 P2P Policy management

P2P policy management is a proprietary solution by Sandvine corp. [14] that attempts to interact directly with the P2P overlay network in order to manage this network according to a policy under the network operator's control. Such a scheme attempts to bridge the gap between the P2P overlay network and the physical topology in order to dramatically reduce the inefficiency present in the uncontrolled system.

This approach is actually a combination of techniques used in traffic shaping and tiered service provisioning: identification of P2P traffic, QoS management of P2P traffic, and deployment of underlying network topology aware applications that act as a facilitator of P2P conversations.

Table 1. Summary of P2P control methods.

Method	**Comments**
Traffic blocking	Effectively stops all known P2P applications. May lead to user dissatisfaction.
Traffic shaping	With the ability to prioritize user traffic, the network operator can control P2P traffic. To do this, positive identification of P2P traffic is required.
Rate limiting	Widely used method in the industry. Does not solve the problem of overlay mismatch and 'passive' traffic.
Over-provisioning and topology upgrade	Aimed at creating more bandwidth. A very expensive approach which doesn't solve the fundamental problem.
Tiered services	An ambitious approach to creating a broadband-friendly P2P application. Problem of popularity amongst users.
Caching	Caching reduces downstream traffic by providing local copies of content. The cache operators risk copyright infringement lawsuits launched against them.
P2P Policy management	Proposes to bridge the gap between P2P overlay and the underlying physical network. Requires positive P2P traffic identification.

1.7 P2P traffic identification

In order to implement several of the above mentioned P2P control methods, it is essential to be able to positively identify P2P traffic. The P2P development community uses several techniques to evade detection such as port-hopping and the use of encryption. These evasion techniques limit the use of the P2P identification methods.

1.7.1 Content inspection

The approach is based on inspecting the contents of packets in an attempt to detect characteristic patterns of P2P protocols. The first thing to do is to infer such patterns, or signatures, from known P2P traffic. A list of signatures is built for all the P2P protocols to be detected. Each packet is then inspected to check if it matches one of the signatures. A study showed that intrusion detection systems IDS can be configured to detect P2P traffic on firewall machines [15]. Recently, content inspection platforms have been built for ISP use. For example, the P-Cube service control platform [16] supports on-the-fly content inspection at the application level. The main limitations of this method are [2]:

- Encrypted traffic can not be inspected and may avoid detection (but at the moment only FastTrack signaling traffic is encrypted, data transfers which account for the P2P traffic volume remains detectable).
- Signatures are rather volatile in nature and must be updated regularly with the evolution of P2P protocols.
- Content inspection at the application level is resource consuming and makes the realization of very high speed switching devices expensive.

The last two points indicate that the access network operators' costs will be increased.

1.7.2 Netflow

Another study outlined the use of Cisco's Netflow services to identify P2P traffic [2]. Netflow is a three-tiered architecture comprised of data export from a routing device, data collection, and data analysis. Once data has been captured and stored, several traffic analysis tools analyze it. The main advantage of this approach is that it doesn't require knowledge about the P2P protocol higher than the transport layer.

The disadvantages are:

- NetFlow traces can represent a huge amount of data thereby requiring the processing of a large amount of traces (data) for P2P traffic detection, with the cost and performance considerations this implies.
- Netflow records are aggregated. The flow abstraction provided by NetFlow obscures details of intra-record exchanges. This makes it difficult to compute packet sizes distribution (which could characterize P2P traffic) or detect special size packets (signaling protocols could use fixed size messages for queries, answers or acknowledgements).
- There is no guarantee of seeing the complete flow of traffic. Depending on the location of data capture, a request may be seen but not responses and vice-versa.

In addition to P2P detection methods, an approach to measure the traffic of a P2P system based on using crawlers is suggested in [2]. A crawler is a client of a P2P system whose sole purpose is to gather statistics about the system. Its main limitation is that this method is very intrusive in nature.

Positive identification of P2P traffic is one of the main tasks to be addressed in this thesis work. Criteria by which P2P traffic can be identified will be proposed. Some identification metrics could be: high traffic intensity and duration of the session.

CHAPTER 2. PUBLIC ETHERNET ACCESS BROADBAND NETWORKS

2.1 Overview of the Public broadband Ethernet

Ethernet is emerging as a standard access technology for broadband networks due to its simplicity of deployment and price-to-performance ratio. Ericsson has come up with a public Ethernet based broadband solution that is scalable and robust in design. The design is described below and the architecture is shown in figure 10.

2.2 Structure of the Public broadband Ethernet

The major components of the public broadband network are depicted below:

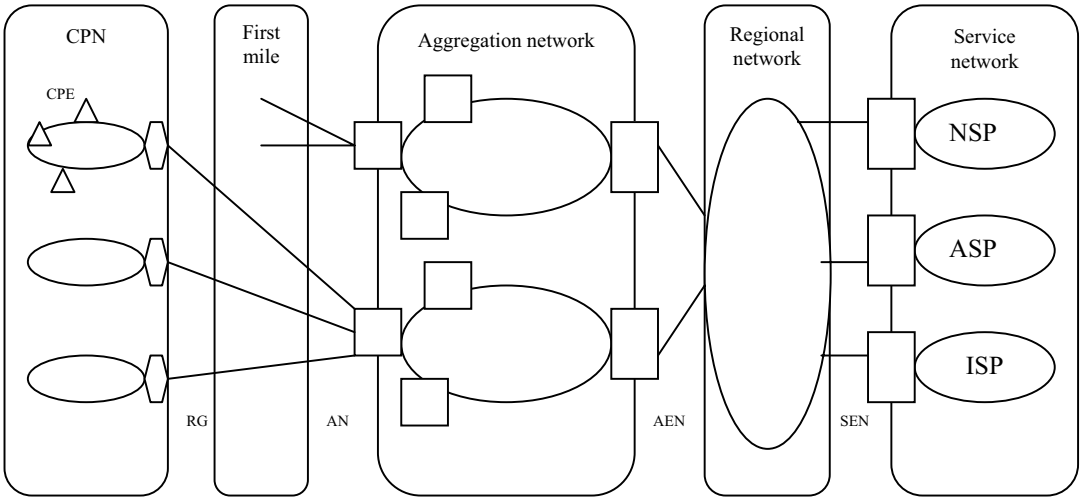


Figure 10. Public broadband access network structure [52]

- CPN (Customer premises network): It consists of customer premises equipment (CPE) connected via digital residential gateways (DRG). The CPN can be a hybrid of different technologies (WLAN, phone line wiring or Ethernet cabling) and is controlled by the user.
- First mile: The physical link connection between the DRG and the Access Node, it can be a DSL, UTP cat5, fiber or wireless connection.
- Access nodes (AN): These are Ethernet switches or DSLAMs depending on the technology used in the first mile
- Aggregation network: Consists of a hierarchy of aggregation switches. It aggregates traffic from first mile to the regional network. The aggregation network and the first mile are collectively called the access network.
- Access edge node (AEN): Also called edge node. It provides security and QoS support
- Regional Network: Usually an optional network. Provides connectivity between the access network and the service networks.
- Service Network: The Service Network encompasses a number of service provider networks and nodes, each offering one or more services. These services

are envisioned to be mainly IP based. It can be run by a Network Service Provider (NSP), an Internet Service Provider (ISP) or an Application Service Provider (ASP). Connected to the regional network by service edge nodes (SEN).

2.3 Public Ethernet broadband requirements

A public broadband access network needs to support traffic separation, service differentiation (quality of service), security, multicast, be robust (in-service performance), and have a telecommunications management solution to support operation and maintenance of the network. Traffic separation prevents end-users from eavesdropping upon the traffic of other end users. It also separates services and other service provider traffic, giving the network operator full control of who talks to whom, thereby guaranteeing that only authenticated users may use network resources. The definition of different classes of quality of service (QoS) makes it possible to differentiate between services—for example, those that are sensitive to delay and packet loss and those that are not. This ensures that the most sensitive applications and the most profitable services receive priority when there is congestion in the network. Congestion may occur due to over-subscription of links. Although most end-users are well behaved, a small percentage of them can be malicious. Therefore, to avoid fraud and service outage, operators must put security mechanisms in place to protect the network and other end users.

2.4 Traffic separation

A characteristic of Ethernet based access networks is that all end user devices in the broadcast domain will be able to send traffic to each other using frames labeled with their MAC addresses, i.e. they can all ‘see’ each other. Another characteristic is that packets with a yet unlearned MAC destination address will be forwarded to all switch ports. In a LAN these characteristics are desirable, but they present several security threats in public Ethernet based access networks. Traffic separation techniques hide the true MAC addresses of end users, this way there is no direct layer-2 visibility between host machines. Forced forwarding techniques can be used to enhance the security. Two alternatives of forced forwarding are described below namely MAC forced forwarding (MAC FF) and McCircuit (McC).

2.4.1 MAC Forced Forwarding

A scheme to prevent direct layer-2 connectivity between users is the MAC forced forwarding method (MAC FF). MAC FF forces all upstream traffic to go through an edge node where security, QoS, and billing policies can be applied. Hence all the service provider policies: billing, accounting, and security, are implemented at this edge node. To make sure user traffic adheres to these policies the MAC FF mechanism forces all traffic from the hosts connected to the access node to go via the edge node. This is implemented by replying to all ARP requests of the clients with the MAC address of the edge node. The access node drops all packets with a destination address other than that of the edge node. This way even if the clients are in the same IP subnet, their traffic is still forced to go via the edge node.

2.4.2 McCircuit

McCircuit is a scalable traffic separation technique that allows the re-use of most of the existing Ethernet based equipment. McCircuit provides a framework to establish, activate, and deactivate service connections that can be used to carry unicast and multicast Ethernet service connections [29]. The service connections are static or semi-static in nature and are created when an end-user subscribes to a service, but the attributes of a particular service connection can be dynamically changed during the process of service connection activation. In McCircuit the service connections are identified by locally administered MAC addresses called McCircuit address.

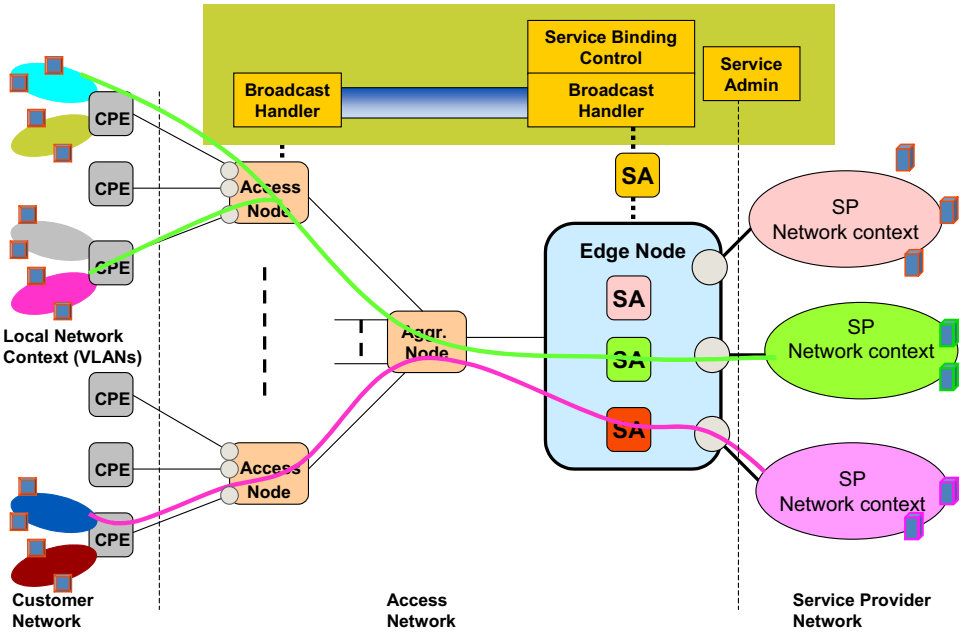


Figure 11. Components of McCircuit based Public access network. [29]

The user establishes a service connection with their respective service provider. A McCircuit address is maintained for the service connection and used as the source address for all user downstream traffic and as the destination address for all user upstream traffic. The broadcast handler tunnels all user broadcast messages to the Edge Node during address configuration in the initial stages of service connection establishment.

2.5. Network technologies of Public Ethernet broadband

The current trend in public broadband networks is the adoption of Ethernet as the access technology in the 'first mile' of the network. An advantage of an access architecture based on Ethernet and IP is to benefit from the volume of components in the LAN market and achieving highly efficient packet based network services plus easy connectivity to user equipment.

2.5.1. Ethernet

Here the term Ethernet refers to the family of local area network products defined by the IEEE 802.3 standard. The standard supports data rates of 10Mbps, 100Mbps, and 1000Mbps over copper and fiber lines.

2.5.1.1. IEEE 802.3

The IEEE 802.2 standard defines a basic data frame format that is required for all MAC implementations, plus several additional optional formats that are used to extend the protocol's basic capability. The basic data frame format contains the seven fields shown in Figure 12.

•Preamble (PRE)—Consists of 7 bytes. The preamble is an alternating pattern of ones and zeros that tells receiving stations that a frame is coming, and that provides a means to synchronize the frame-reception portions of receiving physical layers with the incoming bit stream.

•Start-of-frame delimiter (SOF)—Consists of 1 byte. The SOF is an alternating pattern of ones and zeros, ending with two consecutive 1-bits indicating that the next bit is the left-most bit in the left-most byte of the destination address.

•Destination address (DA)—Consists of 6 bytes. The DA field identifies which station(s) should receive the frame. The left-most bit in the DA field indicates whether the address is an individual address (indicated by a 0) or a group address (indicated by a 1). The second bit from the left indicates whether the DA is globally administered (indicated by a 0) or locally administered (indicated by a 1). When the left most two bits are 00, then the remaining 46 bits are a uniquely assigned value that identifies a single station. If the high order bit is set then the bottom 16 bits identify a defined group of stations, or all stations on the network.

•Source addresses (SA)—Consists of 6 bytes. The SA field identifies the sending station. The SA is **always** an individual address and the left-most bit in the SA field is always 0.

•Length/Type—Consists of 2 bytes. This field indicates either the number of MAC-client data bytes that are contained in the data field of the frame, or the frame type ID if the frame is assembled using an optional format. If the Length/Type field value is less than or equal to 1500, the number of LLC bytes in the Data field is equal to the Length/Type field value. If the Length/Type field value is greater than 1536, the frame is an optional type frame, and the Length/Type field value identifies the particular type of frame being sent or received.

•Data—Is a sequence of n bytes of any value, where n is less than or equal to 1500. If the length of the Data field is less than 46, the Data field must be extended by adding a filler (a pad) sufficient to bring the Data field length to 46 bytes.

•Frame check sequence (FCS)—Consists of 4 bytes. This sequence contains a 32-bit cyclic redundancy check (CRC) value, which is created by the sending MAC and is recalculated by the receiving MAC to check for damaged frames. The FCS is generated over the DA, SA, Length/Type, and Data fields.

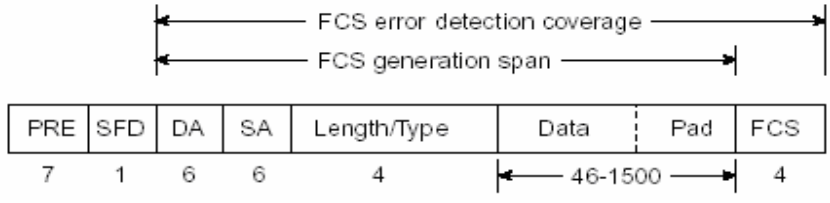


Figure 12. IEEE 802.3 MAC data frame format

2.5.1.2. 802.1Q

IEEE 802.1Q defines Virtual LANs (VLANs). VLANS can be viewed as a group of devices on different physical LAN segments which can communicate with each other as if they were all on the same physical LAN segment. VLANs are defined on the LLC layer by VLAN tags. VLAN tagging is a MAC option that provides three important capabilities:

- A means to expedite time-critical network traffic by setting transmission priorities for outgoing frames.
- Allows stations to be assigned to logical groups, to communicate across multiple LANs as though they were on a single LAN. Bridges and switches filter destination addresses and forward VLAN frames only to ports that serve the VLAN to which the traffic belongs.
- Simplifies network management and makes adds, moves, and changes easier to administer.

A VLAN-tagged frame is simply a basic MAC data frame that has had a 4-byte VLAN header inserted between the SA and Length/Type fields, as shown in Figure 13.

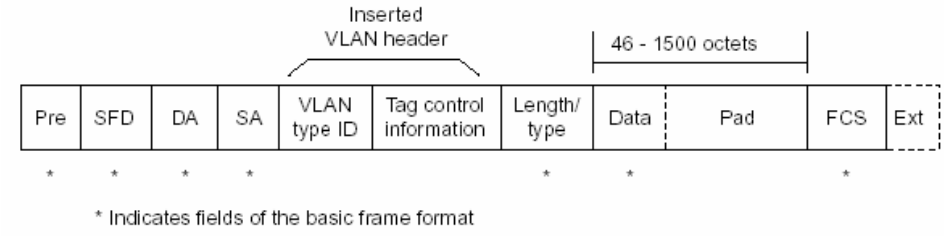


Figure 13. 802.3ac MAC data frame format

The VLAN header consists of two fields:

- A reserved 2-byte type value, indicating that the frame is a VLAN frame
- A two-byte Tag-Control field that contains both the transmission priority (0 to 7, where 7 is the highest) and a VLAN ID that identifies the particular VLAN over which the frame is to be sent

2.5.2. IP

The Internet Protocol (IP) is a network-layer (Layer 3) protocol that contains network addressing information and some control information that enables packets to be routed. IP is documented in RFC 791 and is the primary network-layer protocol in the Internet protocol suite. IP represents the heart of the Internet protocols. IP has two primary responsibilities: providing connectionless, best-effort delivery of datagrams through an internetwork, and providing fragmentation and reassembly of datagrams to support data links with different maximum-transmission unit (MTU) sizes.

2.5.2.1. Header format

An IP packet contains several types of information, as illustrated in figure 14 shows the fields comprising an IP packet.

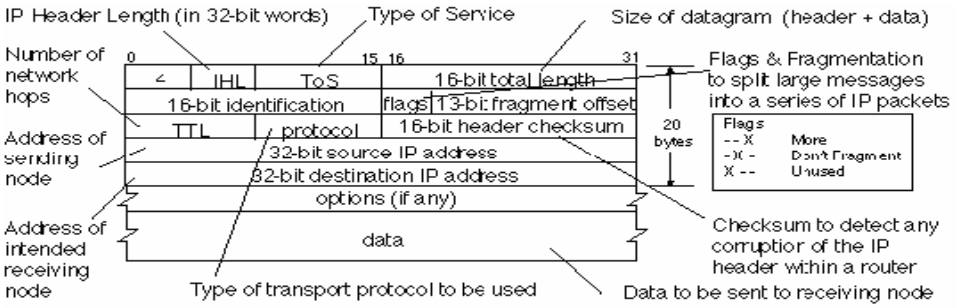


Figure 14. IP packet format

- IP packet fields description:
- Version—Indicates the version of IP currently used (IPv4 = 4, IPv6 = 6).
- IP Header Length (IHL)—Indicates the datagram header length in 32-bit words.
- Type-of-Service—Specifies how an upper-layer protocol would like this datagram to be handled.
- Total Length—Specifies the length, in bytes, of the entire IP packet, including the data and header.
- Identification—Contains an integer that identifies the current datagram. This field is used to help piece together datagram fragments.
- Flags—Consists of a 3-bit field of which the two low-order (least-significant) bits control fragmentation. The low-order bit specifies whether the packet can be fragmented. The middle bit specifies whether the packet is the last fragment in a series of fragmented packets. The third or high-order bit is not used.
- Fragment Offset—Indicates the position of the fragment's data relative to the beginning of the data in the original datagram, which allows the destination IP process to properly reconstruct the original datagram.
- Time-to-Live—Maintains a counter that decrements down to zero at each hop, at which point the datagram is discarded. This keeps packets from looping endlessly.
- Protocol—Indicates which upper-layer protocol receives incoming packets after IP processing is complete.
- Header Checksum—Helps ensure IP header integrity.
- Source Address—Specifies the sending node.
- Destination Address—Specifies the destination node.
- Options—Allows IP to support various options.
- Data—Contains upper-layer information.

2.6 Access Node

The ELN 220 is a member in the Ericsson AXC105 family of Ethernet access products. This switch is designed as a leaf node (Access Node) in an Ethernet access network, i.e. the node that connects to customer premises equipment, for example a Residential Gateway.

2.6.1 Software architecture

The software architecture of the access node relevant for this project is composed of an Erlang processor, the network processor and an interface layer. The network processor is an embedded system that controls the packet switching logic of the device. A set of

interface functions allows the Erlang processor to send and receive data and commands from the network processor.

2.6.2 PAMP

The Penult-Apex Management Protocol (PAMP) version is used to carry information about management and traffic tasks (Penult and Apex are the AN and EN respectively). In version PAMP 1, management task are mainly configuration parameters sent to the AN by the EN. PAMP uses UDP as the transport layer and IP as the network protocol. The PAMP header contains the following fields: Version, Flags, Sequence number, Data length, Data, authentication length, and authentication signature. PAMP commands are acknowledged with an ACK or NACK message. PAMP version 1 contains 12 basic commands as presented in table 2.

Table 2 PAMP command types

Version	Flag	Type	Slogan	Description	Data Length (bytes)	Data
1	Request	3	SETID	Set Penult ID	4	Penult ID
1	Reply	10	ACK	ACK	0	-
1	Reply	11	NACK	Not acknowledged. There was an error in some command. Command processing stops at first command in error.	2	Index of command that was in error.
					Variable	Error code
1	Request	110	ADDMC	Add unicast Service binding.	6	Service binding
1	Request	111	REMMC	Remove unicast Service binding.	6	Service binding
1	Request	114	JOINMMC	Join the specified IPv4 multicast group.	4	Port
					4	Multicast IPv4
1	Request	115	LEAVEMMC	Leave the specified IPv4 multicast group.	4	Port
					4	Multicast IPv4
1	Request	104	SNDETHU	Send the given frame upstream.	Variable	Ethernet frame
1	Request	105	SNDETHD	Send the given frame downstream.	Variable	Ethernet frame
1	Request	106	SNDETHR	Send the given frame back to the port on which it was received.	Variable	Ethernet frame
1	Request	201	SNDBACK	Send the given frame back to the port on which it was received.	Variable	Ethernet frame
1	Request	200	SNDTOPORT	Send the given frame to the given port.	4	Port
					Variable	Ethernet frame

2.6.3 Traffic and control planes

The Erlang processor acts as the control plane, it receives PAMP messages from the edge node and channels them to the network processor with the help of functions provided by the interface library. The network processor controls the switching of ingress and egress packets. It supports several filter modes.

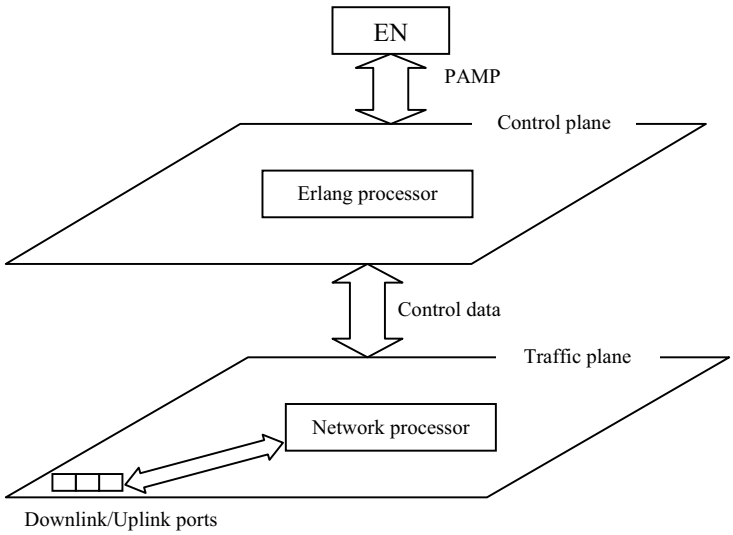


Figure 15. Traffic and control planes of the AN

The NP handles all traffic according to filtering rules written in assembler language. The NP supports several filtering rules. The current filtering rule is set as a parameter during configuration and determines the switching mode of the device. Figure 16 shows the high level logic of algorithm of the NP for packet handling.

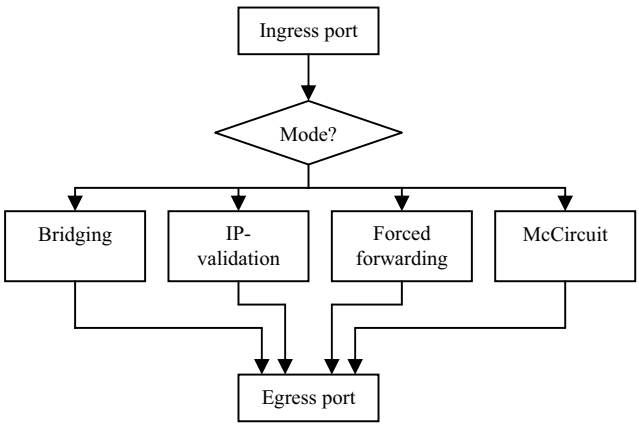


Figure 16. Network processor switching modes

2.7 Measurements of P2P traffic in broadband network access networks

The Phantom project [49] made layer 3 and layer 4 traffic measurements of an access network belonging to a new entrant operator. This section will begin by providing an aggregated picture of the traffic situation in that network. This data will then serve as a base for the analytical model discussed in section 5.2.

Among other measurements conducted, the Phantom project provides information about the traffic volumes exchanged within the network, called the internal traffic, and traffic volumes exchanged between the broadband network and the Internet, this traffic is

referred to as the external traffic. The first graph shows the volume of internal traffic vs. external traffic.

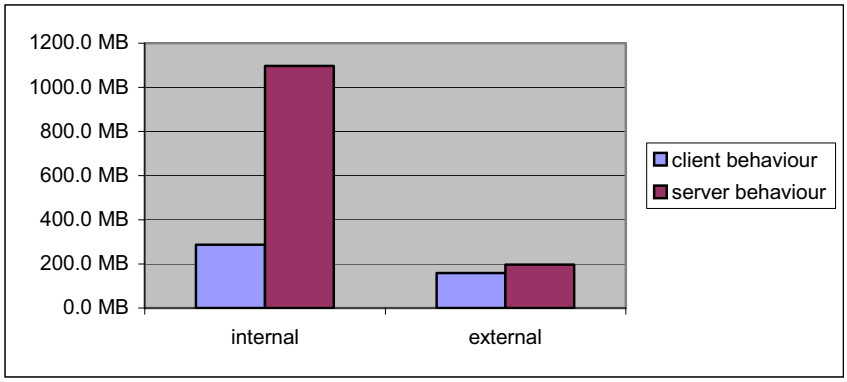


Figure 17. Internal and external traffic for different user behavior [49]

From the figure 17, it is apparent that the traffic generated by users internal to the network is significantly greater than traffic generated between the network and the Internet. However, the next set of data reveals that for the case of unclassified traffic (with respect to the measurements made), the amount of external traffic is greater than the internally originated.

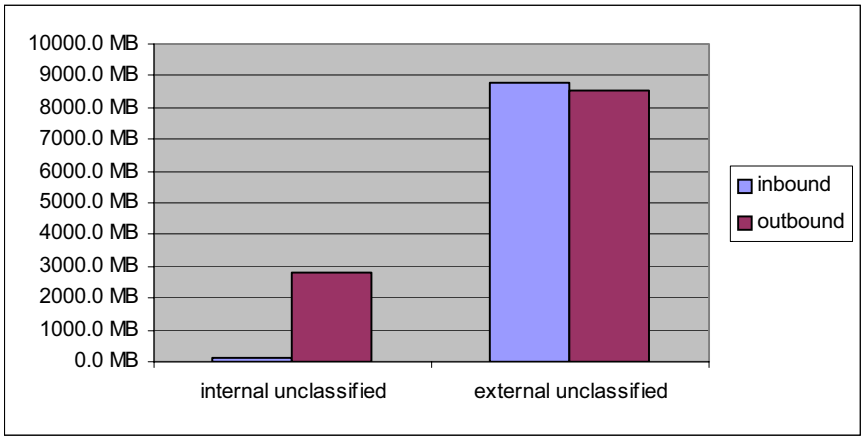


Figure 18. Ratio of internal to external unclassified traffic [49]

The traffic measurements indicate that the unclassified traffic (traffic generated by applications using ports from the undefined range) has a high probability of being traffic generated by P2P applications using non-standard ports. A more detailed view of different types of applications that generate the data of figure 18 is presented in figures 19 and 20.

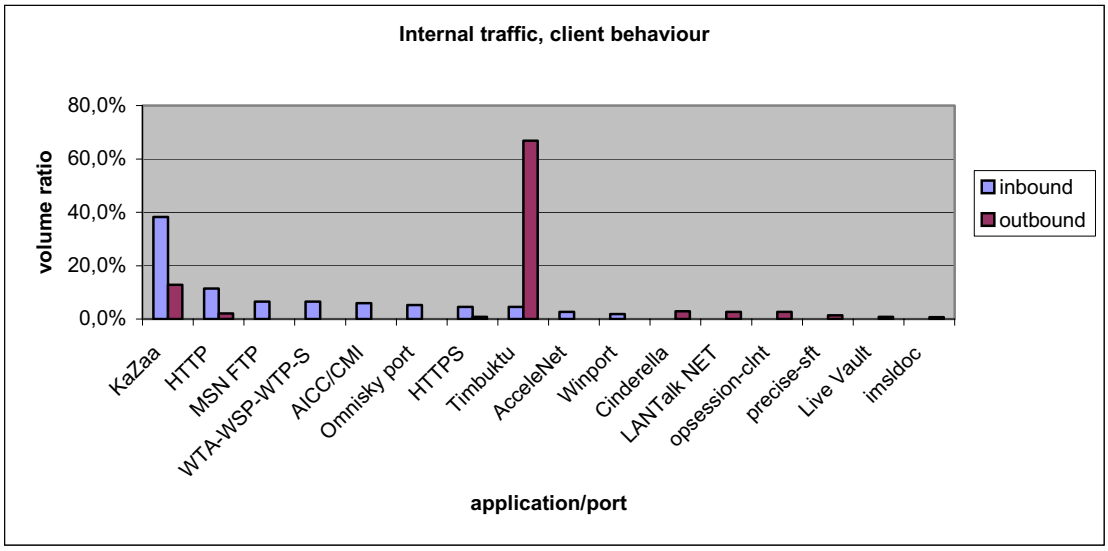


Figure 19. Application composition of internal traffic generated by users acting as clients [49]

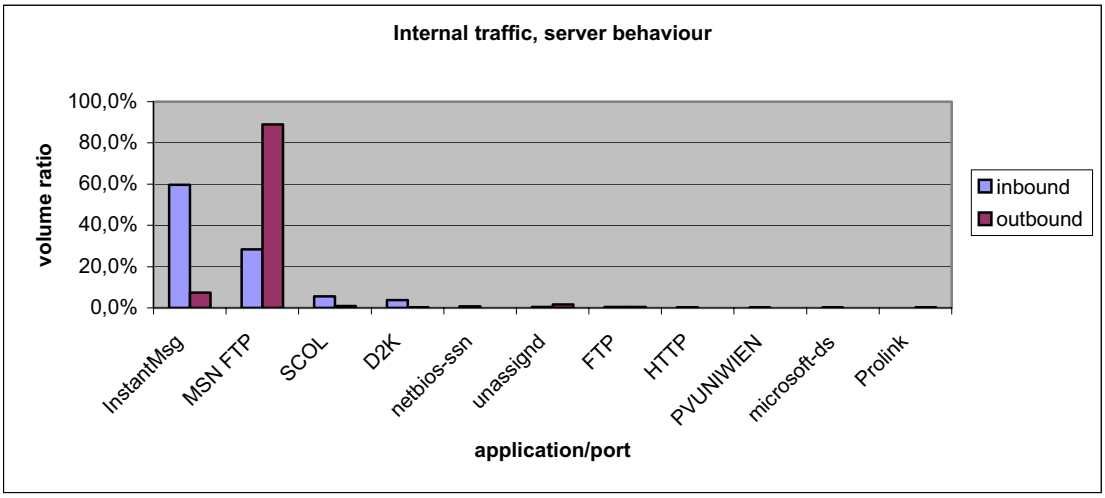


Figure 20. Application composition of internal traffic generated by users acting as servers [49]

Figures 19 and 20 show that P2P applications (specifically, Kazaa and Timbuku) are the greatest contributors to P2P traffic generated in client and server mode. File transfer by FTP and MSN FTP also contributes to the traffic.

The conclusion that can be drawn from these statistics is that the internal traffic generated within the broadband network is significant. The magnitude of this traffic surpasses that of external traffic for the identified applications. P2P applications constitute a large percentage of internal traffic. The measurements of unclassified traffic show that the volume of internal outbound traffic (figure 19) is large. All these indications point to a potential scenario where the internal traffic caused by P2P applications could lead to congestion in the links of the aggregation network. Such a scenario is most likely to occur during peak hours, identified as the time period between 15:00 to 18:00. The next section will introduce an approach to redirecting P2P traffic in the aggregation network and in this way reducing the probability of congestion occurrence in a network scenario where McCircuit is used for traffic separation.

CHAPTER 3. BUSINESS MODEL

This chapter would present an overview of the business models employed in broadband networks. Sections 3.1 and 3.2 will present the business roles and pricing schemes as defined by the MUSE project [52]. These two sections will be an overview of the components that will be used to build a business model for P2P traffic in broadband access networks which is presented in section 3.3.

3.1 Business roles

The MUSE project [52] presents a framework of business roles between different entities of a broadband network.

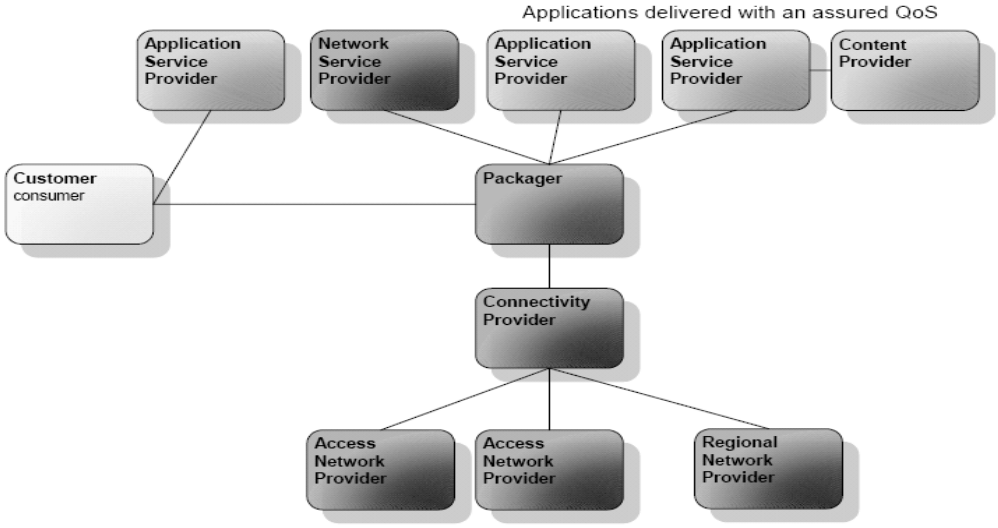


Figure 21. Business Service Roles [52]

The lines between the various roles in figure 21 represent the business relations (in the ideal situation) and not the structure of the network.

3.1.1 Customer

The Customer consumes the services delivered by the service providers. The Customer can be a person or a family, but also a company. If the Customer role is actually fulfilled by a physical person, this person is called the consumer. The consumer could also fulfill other roles in addition to the Customer role.

The Customer has a business relation with the Packager, who may offer him an Internet connection and/or connectivity to a number of Application Service Providers. The Packager acts as the single point of contact to the Customer for network services, but may also be the point of contact for a number of providers.

3.1.1.1 Customer Premises Network (CPN)

The Customer owns a Customer Premises Network or a Company Network. In case of a residential user the Customer Premises Network is connected to the Access Network through a CPE. This device can be simply a bridge but can also have routing functionality

or even functionality that allows dynamic installation of services and applications on the CPE. These kind of CPEs are called a Routing Gateway or Residential Gateway.

3.1.2 Packager

The Packager role has a central place in the business role model. It combines access network functionality from Access Network Providers (ANPs) with core network (Internet, corporate networks) functionality from one or more NSPs or/and application services from one or more ASPs and offers this as a package to the Customer. The Packager is of great value to the Customer as he gives the Customer advice on the modem type and the bandwidth subscription that fits best his/her needs. The Packager is also the single point of contact to Customers with respect to a number of applications that require an assured QoS in the network and in situations where the Customer experiences trouble with a service he has subscribed to via the Packager.

Currently, ISPs are responsible for the NSP role, generally they also fulfill most of the Packager role. The reason for introducing a separate Packager role is to stress the importance of some of the tasks performed by this role. For example, it is envisioned that in situations where the Customer uses services provided by multiple ASPs/NSPs it is necessary to have one (central) actor who has an overview of all the different services (network and application) the Customer has subscribed to, especially when these services require certain QoS settings in the network (access and regional) or CPE.

In general the Packager is technology agnostic. All the technology related aspects of the contract with the Costumer are requirements on the technology specific Connectivity provider. As such the Packager hides all technology from the end user while at the same time is the single point of contact to the Costumer. The Packager may also interface to other Packagers in order to provide nomadic services, ie., a costumer may be able to connect to a “foreign” network if a service agreement exists between the corresponding Packagers enabling them to exchange billing information, user profiles and network requirements. It is the responsibility of the home Packager to convey requirements to the “foreign” Packager in order to support their contract with the Customer.

3.1.2.1 Network Service Provider (NSP)

The Network Service Provider enables Customers to connect to the Internet backbone or a corporate network. Thus, the NSP has SLAs with the Packager of the Customer. The NSP allocates the IP addresses that Customers use to connect with the NSPs network. In the situation that an actor fulfils the NSP role and also has contracts with one or more other actors who fulfill the ASP role, according to the business role model shown in Figure 20 from the Packager/Customer point of view that actor can be considered an ASP as well.

3.1.2.2 Application Service Provider (ASP)

The Application Service Providers manages services above the transport layers. Examples of such services are voice services, a (managed) firewall, video on-demand services, etc. To enable services the Application Service Provider may distribute software that has to be installed on the CPE in the home.
An Application Provider may connect his network directly with the Connectivity Provider’s network (in that case the ASP has a business relationship with the Packager) or he may offer his service via the Internet. Finally, a particular example of an ASP

service could be the delivery and management of a service platform running on the CPE (Residential Gateway). On top of the service platform new applications can be easily installed and started from the network. Third party application providers may deliver the applications.

3.1.2.3 Content Provider

Content Providers make their content, e.g. movies or music, available to Application Service Providers. The ASP provides an end user service with this content by means of a middleware platform which enables the Customer to listen to/watch the content from his end-user-device. Often a Content Provider will have very stringent security conditions to prevent illegal copies of the content before an Application Provider may use this content.

3.1.3 Connectivity Provider

The Connectivity Provider has overall responsibility for providing end-to-end connectivity between the CPE (gateway) and the NSP or ASP network, guaranteeing the agreed QoS and security characteristics. The Connectivity Provider has SLAs with the Access Network Provider and the Regional Network Provider regarding the required network resources. The Connectivity Provider can do authentication and the assignment of IP addresses to CPE on behalf of the NSP or ASP. Further, the Connectivity Provider may assemble billing information from network services and provide this to the Packager.

In general, there will not be more than one Connectivity Provider per CPE, since otherwise it will be hard to control the total amount of bandwidth that a Customer may use. In practice, the connectivity provider role is often combined with the Access Network Provider role or the Regional Network Provider role.

3.1.3.1 Access Network Provider (ANP)

The Access Network Provider is responsible for OSI layer 1 and 2 transport between the CPE and the connectivity provider's edge router. It takes appropriate measures in its network in order to have sufficient resources available to guarantee the agreed QoS. An ANP can offer its network service to multiple Connectivity Providers.

3.1.3.2 Regional Network Provider (RNP)

The Regional Network Provider aggregates traffic from different edge nodes and delivers this to the right NSP or ASP. He may offer his network services to multiple Connectivity Providers.

3.2 Pricing schemes

Several pricing schemes exist [52]. They depend on the parameters that are evaluated by the service provider. A first approach may be:

a. Static Pricing Policy: Charging is independent of the network use.

- Advantage: Easy and cheap implementation.
- Disadvantage: The user QoS is not taken into account. Eg best effort traffic.

b. Dynamic Pricing Policy: The final charge will be based on the traffic flow, the network congestion, the QoS received, etc.

- Advantage: The user has a detailed bill. Favorable for business users.
- Disadvantage: It may be difficult to understand for the customer and expensive to implement by the service operator.

This approach is very simplified. Some, experts have proposed more complex pricing schemes:

1. Volume-Based Pricing: It is related to the amount of data transferred or transmitted/received. It can be based on bytes, packets, connections, and so on.

- Advantage: It provides a means to regulate the total traffic volume and establishes detailed costs.
- Drawback: It will be more expensive to download from a web site with music videos than another that is plain. Another big problem is who pays for the retransmission of packets that are lost or have errors. This will mean more to pay for the same service.

2. Content-based Pricing: It is based on the content and type of the data transmitted (images, music, text, etc). It is quite feasible in a closed network but is difficult (almost impossible) to implement in a global network such as the Internet. Another problem is the lack of a standard that species how charging is to be done. This may cause confusion among users and added costs for the operators.

3. Flat-rate Pricing: A fixed fee will be charged to the user for a given period (monthly, quarterly, etc.).

- Advantage: Easy for the users to understand. It is simple and liked by many users. It simplifies a lot billing tasks for the service provider.
- Drawback: It may be inefficient for the provider. So, it can be combined with techniques that establish a maximum use. If the user goes beyond that limit, an additional fee will be charged. Flat rates are mostly used for best effort internet access, a problem with this is that it's difficult to guarantee a QoS scheme.

4. Paris Metro Pricing (PMP): It divides the network in a set of logical sub-networks with different prices. Each sub-network provides only best-effort services. The most expensive ones will have less congestion than the cheapest.

- Advantage: Provides congestion control for free, once the pricing mechanism is in place, with only minor changes to the network infrastructure being required to handle the traffic management tasks.
- Drawback: It may present some difficulties to be understood by the user. In addition, there is high billing complexity

5. Priority Pricing: This kind of pricing model will guarantee a better QoS. Each packet is marked with a given priority level. So, when the network is congested, the packets are thrown away according to their priority level.

- Advantage: The provider is able to make more efficient use of his network and he can provide different QoS types. Congestion can be prevented with appropriate pricing.
- Drawback: QoS is improved but it is not guaranteed. It may present some difficulties to be understood by the user.

6. Smart-Market Pricing: Users will select resources by adding a packet header. The network gateway will evaluate each packet and will send it according to its header.

- Advantage: The provider is able to make a more efficient use of his network and he can provide different QoS types.
- Drawback: It may present some difficulties to be understood by the user.

7. Proportional Fairness Pricing: It tries to incorporate fairness into resource allocation. Every customer is allocated some bandwidth proportional to his willingness to pay. Model assumes a single path for each user, and then maximizes the sum of utility of all users while respecting capacity constraints.

- Advantage: The provider is able to make a more efficient use of his network and he can provide different QoS types.
- Disadvantage: It is not useful for some applications. It may present some difficulties to be understood by the user and to be implemented by the provider.

8. Edge Pricing: The congestion along the path between emitter and receiver will be calculated and users will be charged according to that congestion level in a given period of time.

- Advantage: The provider is able to make a more efficient use of his network and he is able to avoid hard congestion risks.
- Drawback: It is quite complicated. Utility functions are hard to know, and change over short time intervals. It also may present some difficulties to be understood by the user and to be implemented by the provider.

9. Responsive Pricing: It is a very dynamic methodology. It is based on congestion control. Prices will rise as congestion increases and prices will decrease as congestion eases.

- Advantage: This scheme reduces or eliminates packet drops
- Disadvantage: it improves, but does not guarantee, QoS. It is difficult to be understood by the end user and to be implemented by the provider.

10. Effective Bandwidth Pricing: Effective bandwidth is the required bandwidth of a session. For a “real-time” session, it is the peak bandwidth. But, for a non-real-time session with unlimited buffering, it is the mean bandwidth. Charging in proportion to the mean rate could be implemented when there is a large degree of multiplexing [62]. Users will establish mean and peak bandwidth desired use (traffic profile) during call admission control (CAC) and provider will charge according to those wishes.

- Advantage: Users can change their traffic profile whenever they want to. Charges are based on time and volume, so it is easy to be understood by users.
- Drawback: it improves, but does not guarantee, QoS. Users should be able to understand traffic bandwidth features of their networks before contracting a provider.

11. Location based Pricing: Pricing rates are based on the user location. This scheme is coupled with the nomadic concept. Thus, when a user is near home, it will be easier for the provider to supply the agreed bit-rate. But, if the user is at the airport, in a car or in a department store, more has to be paid for the same bit-rate.

- Advantage: The provider will get more benefits when he is supplying nomadic applications.
- Drawback: It is difficult to provide and to manage this kind of pricing scheme. It needs additional technology, such as Geographic Information System (GIS), and it also needs additional information about the user location. Easily understood by the user.

While each pricing scheme has its pros and cons at the moment, particularly in Northern Europe, operators tend to employ the flat rate pricing scheme. It is argued that its main advantage to the customers is its simplicity. To the operators, a flat rate scheme helps keep down the cost of a billing and accounting system that would otherwise have to be maintained.

3.3 A P2P business model for broadband networks

As the previous section discussed, adopting a pricing scheme where users are billed by the volume of traffic may seem the logical way to tap revenue of P2P. However, this is not a popular solution and may lead to loss of customers. Billing per traffic volume also has additional costs to the network service operator; specifically setting up and maintaining the billing system.

Flat rate is currently a widely deployed pricing scheme. Network service providers are inclined to keep using it for its simplicity and customer preference. The following section will present a business model, which permits the NSP and the ANP to earn revenue from P2P while at the same time the customer pays a flat rate fee for best effort Internet access. The proposed business model is aimed at producing revenue from P2P traffic for the NSP, which would in turn pay the ANP for providing such a feature. A reality to come to terms with is that neither the NSP nor the ANP can change user behaviour; users will continue to use P2P applications and generate high volume of traffic. However the NSP, using traffic-engineering techniques offered by the ANP, can change the path of P2P traffic flow within their networks. The aim is to localize P2P traffic within the ANP; by doing so the ANP has control over P2P traffic and can offer this as a feature to the NSP. The NSP can then start earning revenue by making P2P a chargeable service and at the same time keeping the flat rate pricing scheme for best effort Internet access. From the customers' perspective, the main interest is finding the desired content and downloading it as fast as possible, the origin of the content is of no importance.

There are two types of P2P traffic flows - *internal P2P* and *external P2P* traffic. Internal P2P traffic is the traffic that is generated between users connected to the access network, while the external is the traffic between a user in the access network and the Internet. As measurements show, the external P2P traffic is usually much larger than the internal. There is no way other than charging per volume to extract revenue from external traffic. However, revenue could be extracted from internal P2P by making it an additional

service. So the key to P2P revenue in this business model is encouraging internal over external P2P traffic.

Assuming that P2P objects follow a Zipf-like distribution, it is logical to assume that end users attached to a large broadband network (tens of thousands users) will at any point in time contain the most popular and often downloaded objects. The end users attached to broadband network could then be viewed as a content cache, where the cache is not centralized but distributed on the user nodes. Once a new object is on some node in the network, it is subsequently distributed among the other users locally.

3.3.1 Business relationships

The MUSE project [52] suggests several business models built out of different combinations of relationships between the different business roles and entities. In regard to P2P traffic a three-actor business model consisting of the relationships between the packer, the connectivity provider, and customer is proposed with emphasis on the NSP and ANP roles.

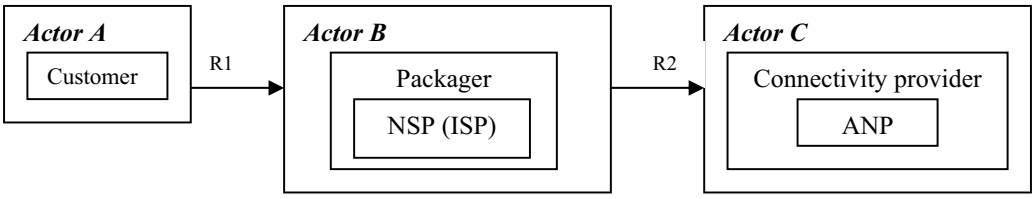


Figure 22. P2P business model

The model assumes the following business relations:

3.3.1.1 R1

The business relationship (R1) between the user and the Packager includes the subscription and the service level agreements (SLA). Users cannot be directly banned from generating external P2P traffic, however, they could be influenced to access local copies. The following pricing scheme is proposed for this:

1. Offer best effort traffic at a flat rate of Xkr/month. Enforce a cap of YMB. If this is exceeded, start charging Zkr per every extra unit volume.
2. Offer P2P as additional service at Lkr/month (L<X) and allow all internal P2P traffic download and upload *for **free***.

The users in their effort to conserve money would strive to exchange P2P traffic only internally. They could use P2P applications that are aware of underlying physical topology that would enable them to get content from other access network users (like a mobile voice network: calls made internally are cheaper than calls made out of this network). For most effective searching, P2P applications that use the supernode architecture could be used. In such systems all user requests are made to a central node or the supernode. The supernode replies with the address of a peer possessing the required content or forwards the request to another supernode. Popular P2P system that implement the supernode architecture include KazuperNode [8], Direct Connect [51], and Kazaa [40]. The P2P diversion algorithm enabled at the access nodes will ensure that P2P traffic is transported within the respective switch domain in the aggregation network.

An important aspect of this model is that the choice of the volume base in the flat rate offer be made correctly (taking into account the profile of an average non-P2P user). For example Bredsbandbolaget sets this cap at 300GB [50], this figure may need to be reduced for the proposed model to provide the financial incentive for the new service.

3.3.1.2 R2

R2 defines the business relationships between NSP and ANP. The ANP has to be able to localize P2P traffic; this could be done using traffic engineering techniques. This is offered as feature to the NSP in return for a fee. Basically the NSP allows the ANP to do peering of its traffic in the access network. The NSP would have to provide the ANP information on the IP addresses leased to its clients. This information will be used to identify P2P peers by the algorithm proposed in the subsequent chapters of this thesis.

3.3.1.3 User

The user has a new service offering unlimited amount of traffic exchange. For a fraction of the basic subscription fee, the user can get all the content desired without having to worry about exceeding this traffic quota.

3.3.1.4 ANP

The main gain for the ANP in keeping P2P traffic local is that by so doing it reduces the probability of congestion in the bottlenecks of the aggregation network. A potential bottleneck in an access network using the McCircuit principle is the link between the aggregation network and the edge node. The growth of P2P traffic generated by users will not lead to extra expenditures for increasing the performance of the EN since P2P traffic will not flow via the EN. Expanding the capacity of higher-levels to match growth P2P traffic would not be required as traffic would be localized within the lower-level links.

3.3.1.5 NSP

The NSP gets a new source of revenue from the P2P service. Additional, the NSP would gain from not having to deal with P2P traffic external as it would be converted to internal traffic. A huge amount of money will be saved on Internet peering costs. The P2P service rate would act as a service differentiation from other NSPs. It has been suggested that users could buy broadband for the sole purpose of P2P file sharing. Thus a P2P service would attract new customers.

3.3.1.6 Summary

1. The proposed business model retains the flat rate model popular among broadband users and allows the NSP and ANP to gain revenue from P2P traffic.
2. The growth of P2P traffic generated by users will not lead to extra expenditures for increasing the performance of the EN since most P2P traffic will not reach it.
3. A P2P service offered by the network operator could act as a means of attracting new customers.
4. With a P2P localization method in place, P2P traffic is managed and contained within the access network and hence reduces Internet peering costs for the NSP.

CHAPTER 4. PROBLEM STATEMENT

In the scope of this thesis, ***P2P traffic*** will be referred to as the traffic exchanged between hosts connected to the aggregation network. It is assumed that this traffic is generated by P2P file-sharing applications. However, the type of application generating the traffic is not very important, more important is that the source and destination of the traffic is within the broadband access network.

The overview presented in the previous chapters concerning traffic of P2P applications in broadband networks showed that this traffic is one of the major challenges to of best effort traffic performance facing broadband access networks. In particular, the aggregation network through which all traffic is forced could experience congestion during peak hours.

The McCircuit traffic separation mechanism doesn't allow for inter-host communication even though the hosts are connected to the same access node and are in the same IP subnet. P2P traffic of peers residing in the client network passes through the aggregation network and returns back to the customer network!

The synthesis of a P2P business model requires that the ANP implement a method of containing P2P traffic within the access network. Such a feature would be beneficial for the ANP in reducing congestion and bring a new source of revenue for the NSP.

A P2P diversion algorithm implemented in the access node would address the above-mentioned problems. The design goals of the solution include:

1. Identification of P2P traffic based on information that is not affected by P2P stealth techniques such as traffic encryption and port hopping.
2. Enable P2P traffic exchange between users connected to the broadband access network while ensuring compatibility with Ericsson's traffic separation technique based on McCircuit.
3. Utilization of the PAMP architecture for access node- edge node communication
4. The provision of an adequate level of security and user control from the edge node.
5. A platform for other P2P solutions that could be implemented in the edge node. The access node is the enforcer of policies that are passed from the edge node.

CHAPTER 5. P2P DIVERSION ALGORITHM IN ACCESS NETWORKS WITH McCIRCUIT TRAFFIC SEPERATION

The McCircuit concept was introduced to serve as a traffic separation technique to provide users with multi-service access. One of the main concepts of McCircuit is to have all user traffic, regardless of destination, flow via the apex or the edge node (EN) where security and traffic shaping policies are applied.

With the increasing popularity of file sharing using peer-to-peer (P2P) applications, a scenario where the aggregation network becomes congested due to such high volume traffic could take place. An approach allowing direct peer-to-peer communication between users connected to the same access network controlled by an edge node becomes necessary. Such an approach would reduce the amount of peer-to-peer traffic propagating through the aggregation network and hence will decrease the burden on the aggregation network elements as well as on the edge node, freeing network resources for other traffic.

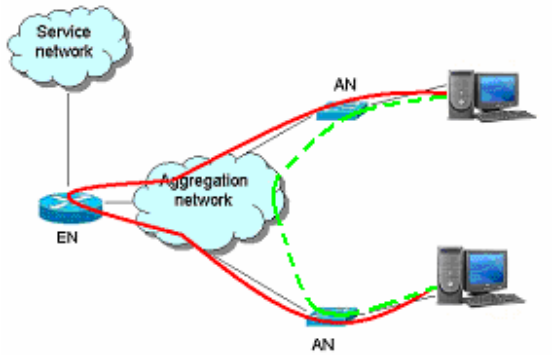


Figure 23. Traffic flow paths with and without P2P looping

In figure 23 the bold line represents the traffic flow path as implemented by McCircuit and the dashed line represents how traffic would flow with the P2P diversion mechanism. The motivation for enabling P2P traffic diversion between users in the customer network is based on the fact that P2P file sharing shows a tendency to match the Zipf-like popularity distribution curve as stated in section 1.3.5. Based on this behavior, it can be inferred that most users' requests will be for a limited set of objects. Once this set of objects is acquired by a user or a group of users in the customer network, there is a high probability that other users will request this set of objects and this way, downloads will end up being made from other users in the customer network.

5.1 Solution overview

The P2P traffic diversion algorithm overrides the McCircuit packet switching rule. This applies only to the packets that have been identified as packets of P2P traffic, all other packets are switched according to McCircuit rules.
The solution focuses on two scenarios for controlling P2P traffic:

- P2P traffic control between hosts connected to the same access node.
- P2P traffic control between hosts connected to different access nodes

Each scenario is composed of the following steps:

- Identification of P2P traffic at the EN
- Building filter table(s) at the access node(s)

- Filtering and modifying P2P packets at the AN

In both scenarios, it is assumed that the users involved in P2P are connected to Internet and thereby have an active service binding characterized by a McCircuit address with the default gateway set as the EN. According to the definition of P2P as given in the problem statement chapter of this thesis, the identification of P2P traffic will be done based on L2 and L3 information. In a nutshell, traffic will be considered ***P2P*** if:

1. The source and destination peers of the traffic are hosts connected to the same access network
2. Additional conditions as specified by policies (section 5.1.3) at the EN are met. These policies are optional.

5.1.1 Hosts connected to a single AN

5.1.1.1 Identification of P2P traffic by the EN

P2P application file sharing session between two hosts starts with a session initiation phase. Host 1 (MAC1, IP1, McC1) sends data to Host 2 (MAC1, IP2, McC2). The EN compares the penult_id in the McCircuit address of the two hosts. If there is a match, meaning that the two hosts are directly connected to the same AN, then the EN creates a table (P2P_table) using PAMP command ADDP2PRW. Prior to the table creation, traffic could be further filtered based on set of traffic policies as described in section 5.1.3.

<SpareM10>	<SA>	<Service|_ID> <Pen|ult_ID>	<Port>

Figure 24. Penult_id and user port in McCircuit header

5.1.1.2 P2P filter table

The table contains traffic identifiers of the participating parties. The table is exported to the AN with the PAMP_ADDP2PRW command. A background process is initialized at the EN, which monitors the service bindings of the two hosts to see if they are active. Once any of the service bindings are terminated, the EN removes the corresponding entries form the P2P_table using the PAMP command DELP2PRW.

Table 3. AN filter table

Check			Modify	
SA	**D.IP**	**s. port**	**DA**	**d.port**
MAC1	IP2	A	MAC2	B
MAC2	IP1	B	MAC1	A

The source ports in the table are the user ports of the access node to which the DRGs are connected. P2P packets are checked to be ingressing the access node from the specified source ports (s.ports), this done to prevent users creating P2P connections at multiple ports of the access node.

5.1.1.3 Filtering and modifying P2P packets

Traffic from host1 and host2 is identified as P2P at the AN by matching the packets' addresses with the P2P_table entries. If there is a match, the traffic is diverted to the port as specified in the destination port (d.port) entry of the P2P table. The destination MAC address of the ingress packets is changed to the source MAC address of the receiving host. As can be seen from table 1, if host 1is sending traffic to host2 as identified by IP2, then the access node changes the destination MAC address of the packets to MAC2 or to

the MAC address of host2. This way traffic will flow directly from host1 to host2 via the AN. A timer is set on every entry in the P2P_table and after a given time period, the entries are aged out. This prevents the P2P connections from remaining for an unlimited period of time. All packets that don't match the P2P_table entries are treated according to McCircuit rules. For example if a malicious user changes the destination MAC address to some existing user's MAC address who is connected to the access node, then the McCircuit switching rule will drop such packets as only packets destined to the edge node are switched to the uplink port. However, MAC addresses could be used to determine the identity of a user. To prevent this, the source MAC address of P2P packets could be change for the respective McCircuit address of the current service binding of the user. This way, the receiving user will see all packets from the peer as originating from the edge node. The dotted field in the figure represents this.

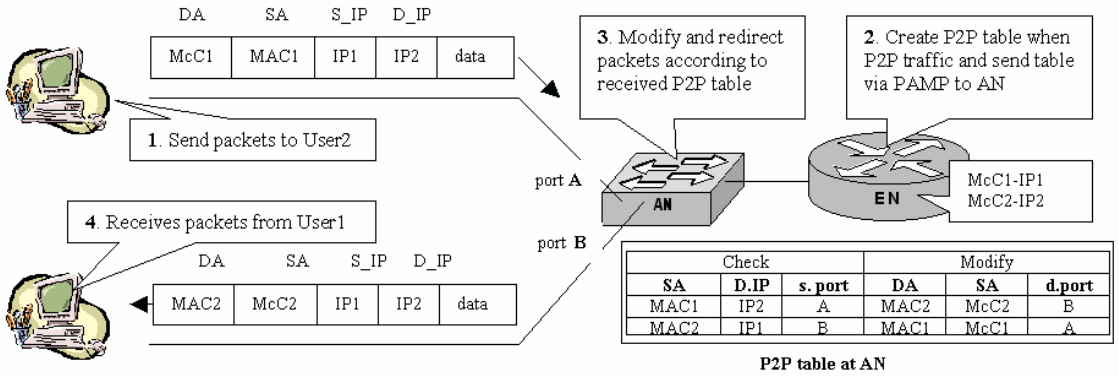


Figure 25. P2PDA: single AN

5.1.2 Hosts connected to multiple ANs

5.1.2.1 Identification of P2P traffic by the EN

The EN maintains an IP address repository (IP_DB); a list of IP addresses with the corresponding MAC and McCircuit addresses leased to the hosts in the customer network. If the penult_ids of the source and destination McCircuit addresses don't match, then the IP addresses are matched against the IP_DB. If the IP addresses of host1 and host2 belong to the access network, then two tables are created and exported to the respective ANs using PAMP command ADDP2PRW.

5.1.2.2 P2P filter tables

The following tables are created and exported to the AN by the EN using the ADDP2PRW command:

Table 4. Access node1 filter table

Check			Modify	
SA	**D.IP**	**s.port**	**DA/SA**	**d.port**
MAC1	IP2	Y	DA: MAC2	uplink
MAC2	IP1	uplink	SA: McC1	Y

Table 5. Access node2 filter table

Check			Modify	
SA	**D.IP**	**s.port**	**DA/SA**	**d.port**
MAC1	IP2	uplink	SA: McC2	X
MAC2	IP1	X	DA: MAC1	uplink

5.1.2.3 Filtering and modifying P2P packets

AN1 changes the MAC destination address of packets of user 1 to the MAC address of user 2 (gray row, table 4). This way the traffic flows via the access network to the access node to which user 2 is connected. At AN2 (gray row, table 5), P2P packets are redirected to port X, the port to which user 2 is connected. Also the source address of the packets is changed to the McCircit address of the service binding of user 2. This is a security measure done to mask the identity of user 1.

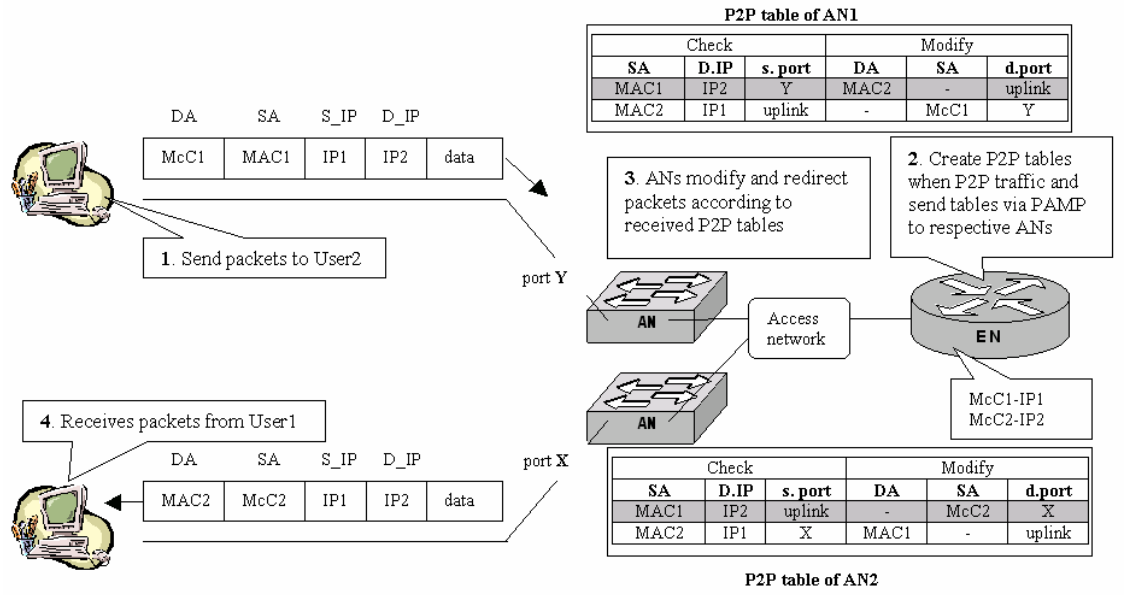


Figure 26. P2PDA: multiple ANs

5.1.3 P2P traffic policies

At the EN policies could be applied to further narrow down the criteria, which would determine what traffic to loop within the aggregation network. These policies give the network operators additional control over the kind of traffic to loop, beyond the fact that both hosts are connected to the access network.

5.1.3.1 Traffic volume

The decision to loop traffic could be based on the amount of traffic volume generated by the flow between the hosts. In this case, the flow should be monitored by the EN for a given period of time and if during this period a threshold value for the amount of traffic (in MBs) has been exceeded, then the traffic should be looped.

5.1.3.2 Traffic type

Another policy for traffic looping could be based on the type of traffic as identified by source and destination TCP ports. For example a rule could be set that HTTP traffic (port 80) and mail (port 25) shouldn't be looped, while ftp (port 20) should be looped.

5.1.3.3 Traffic patterns

If the aim is to loop traffic generated by only a particular set of applications then layer 7 filtering means could be deployed. Hardware or software solutions that implement pattern based recognition algorithms could be used to filter out traffic based on applications, in particular P2P applications.

5.2 Implementation

To support the P2P diversion algorithm the EN and AN would require modifications to be made to their functionality. In the AN the McCircuit logic that is implemented in assembly code would need to be modified. The PAMP protocol would also require add-ons and changes in both the EN and the AN.

5.2.1 AN

5.2.1.1 Switching logic

A change to the code of the network processor to accommodate P2P diversion algorithm has to be made. The P2P logic will be executed prior to the execution of the McCircuit logic. If the ingress packet is a P2P packet then it is sent to the port as indicated in the P2P table, else it's processed by the McCircuit logic. Figure 27 shows this modification.

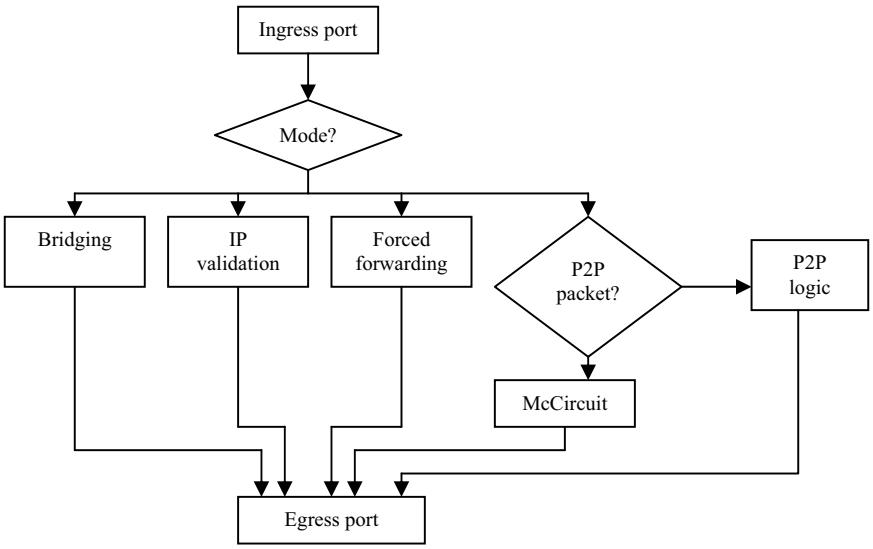


Figure 27. AN switching logic modified for P2P support

5.2.1.2 Filter tables

Currently the AN supports several types of modes, among them is the VMAC (forced forwarding) mode. The following entries are used in the bridging table for VMAC mode: port number, IP, VLAN ID, MAC, VMAC.

The table would have to be expanded to accommodate the destination port column. The following entries are stored in the bridging table: source port, IP, MAC, destination MAC, destination port.

5.2.1.3 P2P PAMP commands

Two new commands were introduced for supporting P2P logic at the access node:

- PAMP_P2PADDRW populates the P2P table in the access node with the traffic identities of the two hosts exchanging P2P traffic.
- PAMP_P2PDELRW removes entries in the P2P table at the access node.

Table 6. PAMP P2P commands

Version	Flags	Type	Slogan	Description	Data length	Data
1	Request	300	*ADDP2PRW*	Adds a row into the P2P table	Variable	SA, s. port, D.IP, DA
1	Request	301	*DELP2PRW*	Deletes a row from the in P2P table	Variable	SA, s. port, DA

5.2.2 EN

As stated in earlier chapters of this thesis, one of the main challenges facing operators is the positive identification of P2P traffic. The approach for detecting P2P traffic within the scope of this work is limited to the identification of the source and destination of traffic in the aggregation network. If both source and destination hosts exchanging traffic are connected to the aggregation network via the access node, then such traffic is said to be P2P. This information can be inferred by looking up the source and destination addresses of packets in the database containing all the currently leased IP addresses. Such a database could be maintained at the EN and updated as users connect and disconnect from the network. The P2P algorithm at the EN contains the following logic:

```
If (S.IP&D.IP belong to EN IP_DB & traffic_policy)
{
        create P2P_table
        export P2P_table to AN using ADDP2PRW
}
If (any user terminates service binding)
{
        delete P2P_table entries using DELP2PPRW
}
```

Apart from identifying P2P packets based on their source and destination addresses, additional policies could be used to specify which packets are to be classified as P2P. A provision for this is made in the algorithm by having an additional match in the conditional statement.

5.2.3 PAMP interaction between AN and EN

The P2P PAMP commands described in the previous section are responsible for maintaining and updating the P2P_table in the EN as well as in the AN. Depending on the events; entries are added or removed from the P2P_table. Tables 7-9 describe the interaction between the AN and EN based on events.

Table 7. Establishing a P2P connection state in AN and EN

	AN	EN
	Start of P2P connection between H1 and H2	

1		IF(P2P traffic)
		{
		Create entry in P2P connection list
		Send: PAMP_ADDP2PRW(H1-H2)
		}

2	Add new entry into P2P_table Repy EN: PAMP_ACK	

Table 8. Service termination by a peer

	AN	EN
1		H1 or H2 terminates service binding
2		Remove [H1-H2] entry from connection list
3		Send: PAMP_DELP2PRW(H1-H2)

4	IF(PAMP_DELP2PRW)
	{
	remove entry from P2P_table
	Reply EN: PAMP_ACK
	}	

Table 9. The AN ages out a P2P entry from its bridging table

	AN	**EN**
1	Ageout timer expires for a P2P connection H1-H2	
2	Entry [H1-H2] removed from P2P_table	
3	Send PAMP message to EN: PAMP_AGEOUT(H1-H2)	
4		Remove P2P entry [H1-H2] from list of P2P connections

5.2.4 Network bottlenecks

To be cost-efficient, access networks are built with a certain degree of over-subscription. Oversubscription means that various services contend for bandwidth, and the use of oversubscription is a given in nearly all networks. However, the network location where the contention takes place varies widely among implementations. Congestion occurs at points of substantial speed mismatch and points of aggregation. Figure 28 shows the possible bottlenecks in access network.

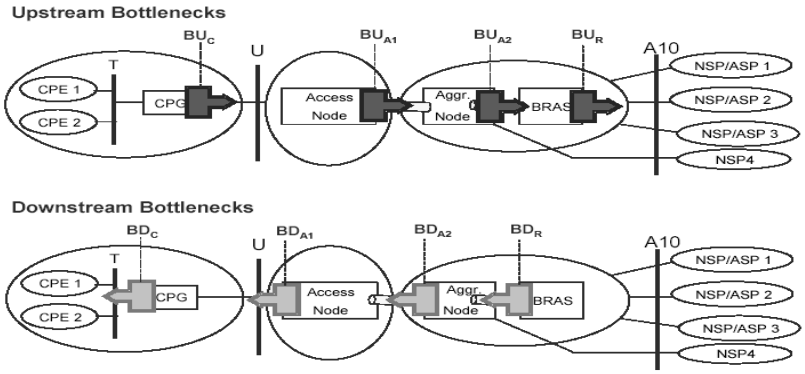


Figure 28. Major bottlenecks [52]

5.2.4.1 L2 QoS

To alleviate bottlenecks and provide users a guaranteed level of service as specified in the SLA different QoS mechanisms are deployed. A common means to classify frames for QoS purposes is their membership in a VLAN. IEEE 802.1Q VLAN tagging provides a standard and interoperable method to indicate frame membership in a VLAN. A distinct QoS service may then be applied on a VLAN-wide basis; that is, to all frames that belong to the same VLAN. VLAN membership could be port based, MAC-based, or by using VLAN tags. Tagging a frame adds a Tag Header as shown in figure 29

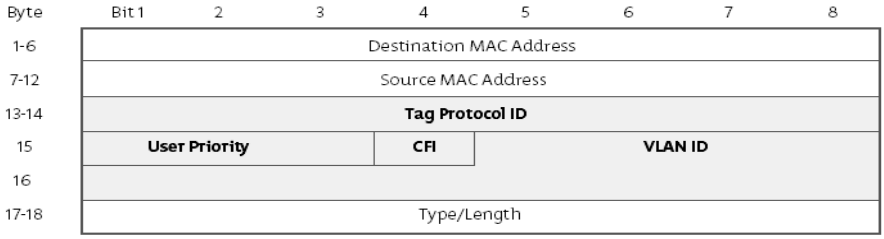


Figure 29. IEEE 802.1Q VLAN Tag and 802.1p User Priority.

The header consists of the following fields:

- Tag Protocol Identifier (TPID), whose value is set as 0x8100
- Tag Control Information (TCI)
- Embedded Routing Information Field (E-RIF), where required.

The 2-byte Tag Control Information (TCI) field consists of a:

- 3-bit User Priority setting, capable of eight priority levels (0 through 7) to allow users to mark the frame for the desired treatment
- 1-bit Canonical Format Indicator (CFI) setting to indicate the format (canonical or non-canonical) of the MAC address
- 12-bit VLAN ID (VID) to identify the VLAN to which the frame belongs.

Another common means to classify frames for QoS purposes is to consider their importance relative to other frames and then to assign them to separate queues based on their relative priority. IEEE 802.1p provides a standard and interoperable way to set the priority bits in a frame's header and to map these settings to traffic classes, with each class corresponding one-to-one to a distinct queue.

Traffic Classes are a means to group frames from several priority settings together and map them to the appropriate queue. Each traffic class corresponds to exactly one queue. A higher numbered class has a higher priority than a lower numbered class, starting with 0 (lowest) and increasing sequentially to 7 (highest).

Table 10. 802.1p QoS priorities

Priority	Description
7	Network Control: 'must get there' requirement
6	Voice: delay < 100 millisecond
5	Video: delay < 10 millisecond
4	Controlled Load
3	Excellent Efforts: or "CEO's best effort"
0	Best efforts
2	Spare
1	Background: bulk transfers

The following mapping table is defined by the standard.

Table 11. IEEE 802.1p User priority and traffic classes.

	Number of available traffic classes							
User priority	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
0 (default)	0	0	0	1	1	1	1	2
1	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	1
3	0	0	0	1	1	2	2	3
4	0	1	1	2	2	3	3	4
5	0	1	1	2	3	4	4	5
6	0	1	2	3	4	5	5	6
7	0	1	2	3	4	5	6	7

If a switch has only one traffic class (meaning, only one queue), then all frames irrespective of priority settings are mapped to traffic class 0 (meaning, queue 0, as shown in column 1 of the table 11). With only one queue, no differentiated services are available and frames are served on a FIFO basis. If a switch has two traffic classes (meaning, two queues), then frames with a priority setting of 3 or lower are assigned to the lower priority traffic class 0 and frames with a priority setting of 4 or higher are assigned to the higher priority traffic class 1 (as shown in column 2).

5.2.5 Sandbox

In this thesis work, a scenario is assumed where the access networks are built based on pre-provisioned “overlay networks”. The overlay consists of pre-provisioning pipes with reserved resources between access nodes and edge nodes. Using 802.1Q in an Ethernet, pre-provisioned pipes can be defined in three ways:
- based on priority bits;
- based on VLAN id’s;
- based on both priority bits and VLANs.

In this thesis the approach of using both VLAN ids and priority bits (p-bits) will be used. VLAN tagging can be used to differentiate between services. Priority bits can be used in addition to VLAN tagging in order to create a finer QoS differentiation.

The following traffic classes are recommended by ITU and 3GPP [58]:

Traffic class		3GPP	ITU
Elastic	Non-interactive	Best-effort	Non-critical
	Interactive	Transactional	Responsive
Inelastic	Non-interactive	Streaming	Timely
	Interactive	Real Time	Interactive

L2 end-to-end QoS pipes for these traffic classes are mapped to service VLANs within the access network. The p-bits are used to set the priority of the traffic based on class. For example real time traffic is assigned priority 6 or 7.

In this traffic classification, P2P would belong to the Best effort class. However, since P2P has the tendency to generate a large amount of traffic, a scenario is possible where during peak hours non-P2P traffic (such as web browsing) is choked out. This would cause user dissatisfaction.

An approach to address this problem is presented by the Sandbox solution developed in this thesis work. The main feature of the Sandbox is the creation of a separate traffic class to which P2P traffic is restricted and the service parameters of which are under the control of the EN. Implementation wise, a P2P VLAN is created. This VLAN runs end to end between all nodes of the access network. To ensure that P2P traffic does not interfere with all other traffic, it is assigned the lowest priority with the priority bit set to 1. All other traffic should be assigned priorities from 0 to 7. The priority bit for best effort traffic is set to 0. Such a change would have a minimum effect on the existing set up or on administration costs. The resulting resource distribution is show in figure 30.

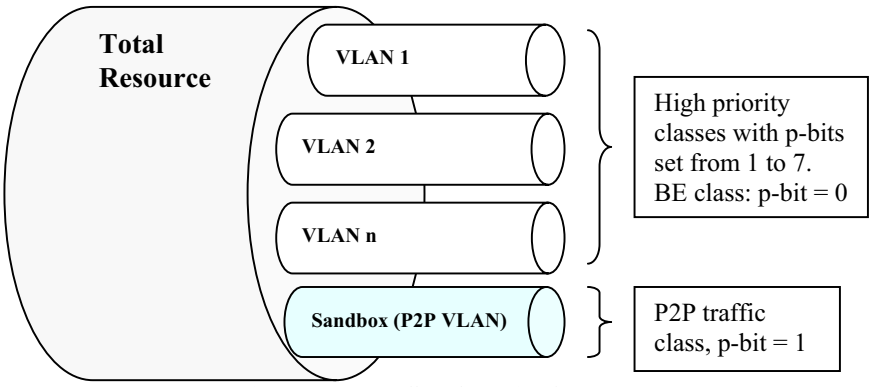


Figure 30. Broadband traffic classes

All uplink ports of the access nodes in the access network are part of the P2P VLAN. Once P2P traffic has been identified by the P2P diversion mechanism, a VLAN tag corresponding to the P2P VLAN is set in the packets and the priority bits of P2P packets are set to 1. The packets are then diverted to the uplink ports and flow to the target access node to which the destination peer is connected. The edge node is not part of the P2P VLAN; this way P2P traffic is not forwarded out of the access network.

During periods of low network usage P2P traffic would be able to use all the existing spare bandwidth, but once other traffic classes appear, P2P traffic will be given less resources. Additionally queue-scheduling algorithms such as the Weighted Fair Queuing (WFQ) or the Weighted Round Ribbon (WRR) could be used. These algorithms influence the rations of bandwidth allocated to different classes.

CHAPTER 6. ANALYTICAL MODEL

The McCircuit concept is based on having all traffic, regardless of destination, be directed towards the edge node for security and for the enforcement of QoS parameters . With the growing popularity of P2P file-sharing applications, a large amount of file sharing traffic is generated between users in a broadband network. For this reason the idea of allowing direct communication between hosts in the aggregation network without having it flow up to the edge node and back is envisioned. As mentioned in the previouse chapter, in the proposed method the AN changes the destination MAC address of the packets of the P2P traffic and this way the traffic flows via the shortest path in the aggregation network to its destination without first flowing via the EN.

To illustrate the gain made by the P2P diversion algorithm, the topology of the aggregation network will be divided into sub-trees called ***switch domains***. Then the intensity created by P2P traffic will be assessed in the uplink of the top switch of each switch domain (the thick dash-dotted links) for the case when the ANs are in McCircuit mode and when in P2P diversion mode. To do this, each switch at the top of a switch domain will be defined in terms of a M/M/1 queuing theory model.

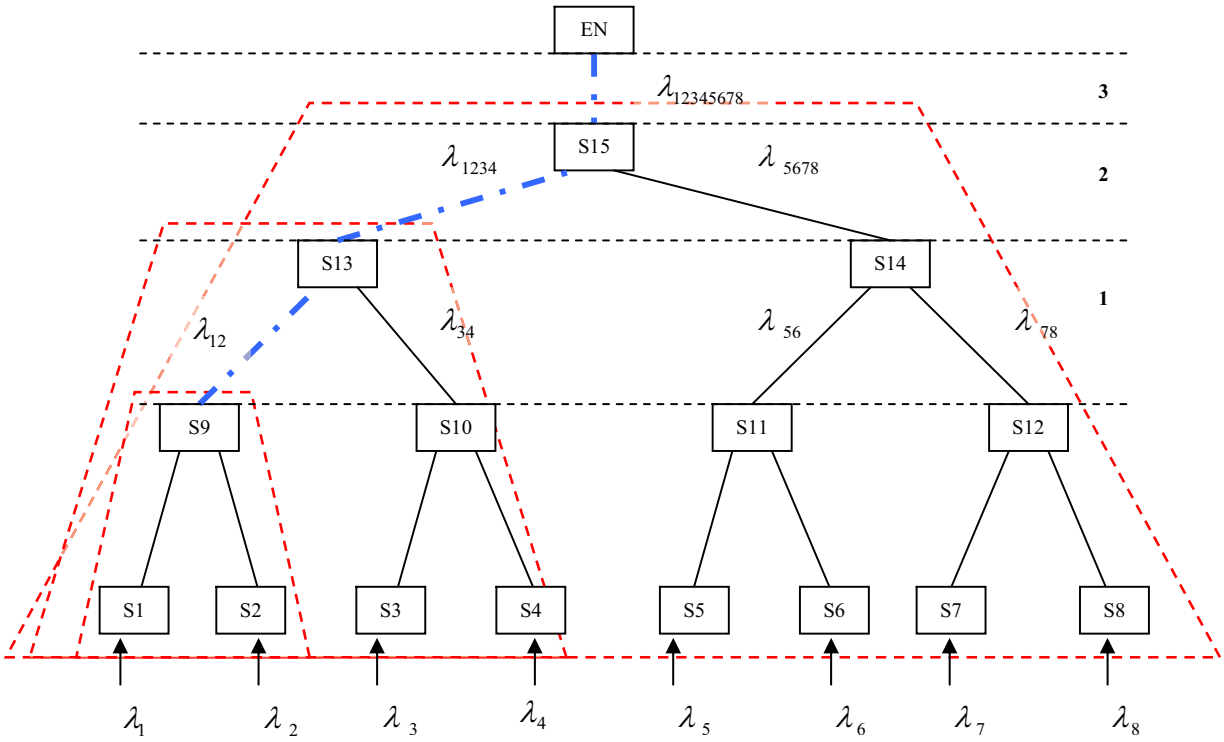


Figure 31. Switch domains of a sample topology

In figure 31, the group of switches enclosed in the dashed boundaries represents a switch domain of the respective level. The switches S1-S8 represent the ANs where the P2P traffic diversion algorithm is executed and where the addresses of P2P traffic packets are modified. End user machines are connected to the ANs via residential gateways (DRGs) and generate traffic with intensities denoted by $\lambda_1 - \lambda_8$. The number of users and their behavior patterns determines the magnitude of the traffic intensity generated.

Model assumption: In the following model a static view of the network is presented; session-level behaviour of users and connection-level behaviour of the applications are not considered.

6.1 Traffic load

The main parameter for evaluating the effect of P2P diversion algorithm (P2PDA) will be the traffic load generated by P2P traffic over the links of the aggregation network.
In figure 31, ρ_{ij} is the load created by the traffic flow of intensity λ_{ij} from switch *i* (Si) to switch *j* (Sj) on the corresponding link. Traffic in link S15-EN signifies the traffic load that would be handled by the service providers connected to the EN each time hosts in the access network communicate. In accordance with McCircuit principle, all traffic must flow following via the EN. The traffic intensities converging at link *ij* are: λ_{ip2p}, λ_{inp2p}, λ_{jp2p} and λ_{jnp2p}. λ_{p2p} denotes the intensity of P2P traffic generated by users connected to the AN. λ_{np2p} denotes the intensity of non-P2P traffic which must flow to the EN. Since non-P2P traffic is not affected by the traffic looping mechanism, λ_{inp2p} and λ_{jnp2p} will be summed to give one value λ_{np2p}. Following the rule of superposition of traffic intensities, the intensity of traffic in link *ij* is given by the sum of the intensities flowing in the link expressed as:

$$\lambda_{ij} = \lambda_{ip2p} + \lambda_{jp2p} + \lambda_{np2p} \qquad (1)$$

Traffic load on network links will be calculated for two modes - McCircuit mode and P2P diversion mode.

Using a M/M/1 model, traffic load is calculated using the following expression:

$$\rho = \frac{\lambda}{\mu} \qquad (2)$$

Where μ - the service rate. Expressing this load in terms of average service time:

$$\rho = \lambda \overline{X} \qquad (3)$$

The inverse value of the service rate is the average packet length:

$$b = \frac{c}{\mu} \qquad (4)$$

From (1) and (3):

$$\rho = \lambda \frac{b}{c} \qquad (5)$$

Where c is the link capacity or speed in units of Mbits/s.

The average packet length may have a different distribution function depending on the traffic type. Packet size distributions in the Internet are trimodal: 40-44 bytes - TCP ACKs, 552 or 576 bytes - Default MSS, when MTU Discovery is not used is 512 or 536 bytes and 1500 bytes MTU for Ethernet. Assuming that the packet lengths of the traffic generated in figure 5 are distributed according to the Exponential distribution then the following expressions are true for the arrival rates.

From the point of view of P2P traffic, the load on link ij can be expressed as:

$$\rho_{ij\_McC} = (\lambda_{ip2p} + \lambda_{jp2p} + \lambda_{np2p})\frac{b}{ci} \quad (6)$$

Where $\lambda_{ip2p}, \lambda_{jp2p}$ are the intensities of P2P traffic of nodes i and j; λ_{np2p} is the non-P2P traffic intensity.

Expression 6 gives the load created by P2P traffic on links in the aggregation network when the ANs are in McCircuit mode.
The effect of P2P traffic diversion will be measured by assessing the traffic intensity in the uplinks of the top switches of switch domains. This intensity will be called the ***exit intensity***. A table showing the expressions for the exit P2P intensities for the case of McCircuit mode and P2P diversion algorithm mode will be presented.

6.1.1 Exit intensity for interaction between single switches

To illustrate the impact of the P2P diversion algorithm (P2PDA) on the links of the access network, P2P traffic exchange between hosts connected to different switch domains will be examined.

Table 12. Case S1->S2 (switch domain S9)

Link	McC	P2PDA
S9-S13	λ_{12}	0
S13-S15	λ_{12}	0
S15-EN	λ_{12}	0

Table 13. Case S1->S3 (switch domain S13)

Link	McC	P2PDA
S9-S13	λ_{13}	λ_{13}
S13-S15	λ_{13}	0
S15-EN	λ_{13}	0

Table 14. Case S1->S8 (switch domain S15)

Link	McC	P2PDA
S9-S13	λ_{18}	λ_{18}
S13-S15	λ_{18}	λ_{18}
S15-EN	λ_{18}	0

In the following section an aggregate expression will be derived for all the possible P2P interaction between peers belonging the different switch domains.

6.1.2 Exit intensity for interaction between switch domains

For the purpose of deriving an expression for all possible connections between hosts connected to the aggregation network, two terms will be introduced:

- P2P_UP traffic intensity (λ_{p2pUP}) is the P2P traffic destined to peers connected to neighboring switch domains. This traffic flows to the uplink port of the aggregation network switch.
- P2P_DOWN traffic intensity ($\lambda_{p2pDOWN}$) is the P2P traffic destined to peers connected to the same switch domain. This is traffic that is diverted by the P2P diversion algorithm. This traffic flows to one of the downlink ports of the switch.

The load for P2P traffic diversion mode can thus be expressed as:

$$\rho_{ij\_p2p} = (\lambda_{i\_p2pUP} + \lambda_{i\_p2pDOWN} + \lambda_{j\_p2pUP} + \lambda_{j\_p2pDOWN} + \lambda_{mp2p})\frac{b}{ci} \qquad (7)$$

To express the intensity of P2P traffic in the links between the switch domains of the topology, the following matrix will be used. The matrix contains the all the possible P2P connections between peers connected to the respective ANs.

$$\begin{bmatrix} \lambda 11 & \lambda 12 & \lambda 13 & \lambda 14 & \lambda 15 & \lambda 16 & \lambda 17 & \lambda 18 \\ \lambda 21 & \lambda 22 & \lambda 23 & \lambda 24 & \lambda 25 & \lambda 26 & \lambda 27 & \lambda 28 \\ \lambda 31 & \lambda 32 & \lambda 33 & \lambda 34 & \lambda 35 & \lambda 36 & \lambda 37 & \lambda 38 \\ \lambda 41 & \lambda 42 & \lambda 43 & \lambda 44 & \lambda 45 & \lambda 46 & \lambda 47 & \lambda 48 \\ \lambda 51 & \lambda 52 & \lambda 53 & \lambda 54 & \lambda 55 & \lambda 56 & \lambda 57 & \lambda 58 \\ \lambda 61 & \lambda 62 & \lambda 63 & \lambda 64 & \lambda 65 & \lambda 66 & \lambda 67 & \lambda 68 \\ \lambda 71 & \lambda 72 & \lambda 73 & \lambda 74 & \lambda 75 & \lambda 76 & \lambda 77 & \lambda 78 \\ \lambda 81 & \lambda 82 & \lambda 83 & \lambda 84 & \lambda 85 & \lambda 86 & \lambda 87 & \lambda 88 \end{bmatrix}$$

The matrix will be used to derive the P2P intensity expressions for the each switch domain.

For switch domain S9:

$$\lambda_{1p2p} = \{\lambda_{12} + \lambda_{13} + \lambda_{14} + \lambda_{15} + \lambda_{16} + \lambda_{17} + \lambda_{18}\}$$
$$\lambda_{2p2p} = \{\lambda_{21} + \lambda_{23} + \lambda_{24} + \lambda_{25} + \lambda_{26} + \lambda_{27} + \lambda_{28}\}$$
$$\lambda_{1-2\_p2p\_DW} = \lambda_{12} + \lambda_{21}$$

Assuming that the traffic intensities between nodes are roughly equal, the P2P traffic intensity for switch S9 can be expressed as a sum:

$$14\lambda_{p2p\_UP} - 2\lambda_{p2p\_DW} \qquad (8)$$

For switch domain S13:

$$\lambda_{12p2p} = \{\lambda_{12} + \lambda_{13} + \lambda_{14} + \lambda_{15} + \lambda_{16} + \lambda_{17} + \lambda_{18} + \lambda_{21} + \lambda_{23} + \lambda_{24} + \lambda_{25} + \lambda_{26} + \lambda_{27} + \lambda_{28}\}$$
$$\lambda_{34p2p} = \{\lambda_{31} + \lambda_{32} + \lambda_{34} + \lambda_{35} + \lambda_{36} + \lambda_{37} + \lambda_{38} + \lambda_{41} + \lambda_{42} + \lambda_{43} + \lambda_{45} + \lambda_{46} + \lambda_{47} + \lambda_{48}\}$$
$$\lambda_{12-34\_p2p\_DW} = \lambda_{12} + \lambda_{13} + \lambda_{14} + \lambda_{21} + \lambda_{23} + \lambda_{24} + \lambda_{31} + \lambda_{32} + \lambda_{34} + \lambda_{41} + \lambda_{42} + \lambda_{43}$$

Summing to:

$$28\lambda_{p2p\_UP} - 12\lambda_{p2p\_DW} \qquad (9)$$

For switch domain S15:

Using a similar approach and assumptions for deriving (8) and (9) the intensity expression for S15:

$$56\lambda_{p2p\_UP} - 56\lambda_{p2p\_DW} \qquad (10)$$

The general expression for the UP and DOWN intensity ratios in a given topology:

$$UPi = Ni(M-1) \qquad (11)$$

$$DOWNi = Ni(Ni-1) \qquad (12)$$

$$M = fr^{L} \qquad (13)$$

Where: N – number of ANs of the ith switch domain,
M – total number of ANs in the topology,
fr – fan out ratio between switches (number of downlink ports occupied by ANs, fr=2 for sample topology presented in figure 31),
L – number of level of switches in the aggregation network.

Table 15. Exit P2P traffic intensities for different switch domains

	Exit P2P intensities	
Level	**McC mode**	**P2P mode**
1	14	14UP-2DW
2	28	28UP-12DW
3	56	56UP-56DW

Different ratios of UP and DOWN P2P traffic will have an impact upon the amount of the exit P2P traffic of a switch. Figure 32 shows how the value of P2P_DOWN affects the exit P2P intensity. The value of P2P_DOWN ranges from 0 to 1.

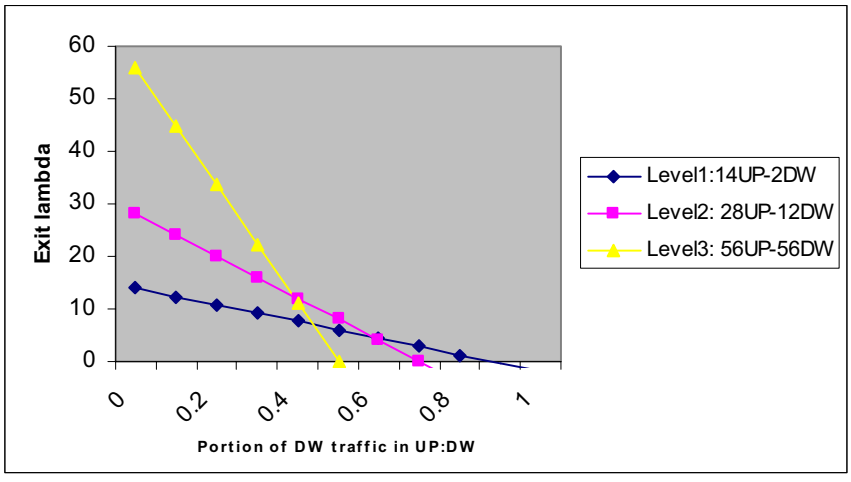


Figure 32 Dependency of exit traffic intensity on portion of P2P_DOWN traffic

Figure 33 illustrates the percentage of traffic that is diverted by the P2P mechanism depending on the percentage of DOWN_P2P traffic. For the top switch of a switch

domain 50% of DOWN_P2P traffic produces a 100% of the total traffic diverted. This means that there will be no P2P traffic flowing to the uplink of the switch if the portion of DOWN_P2P traffic constitutes 50%.

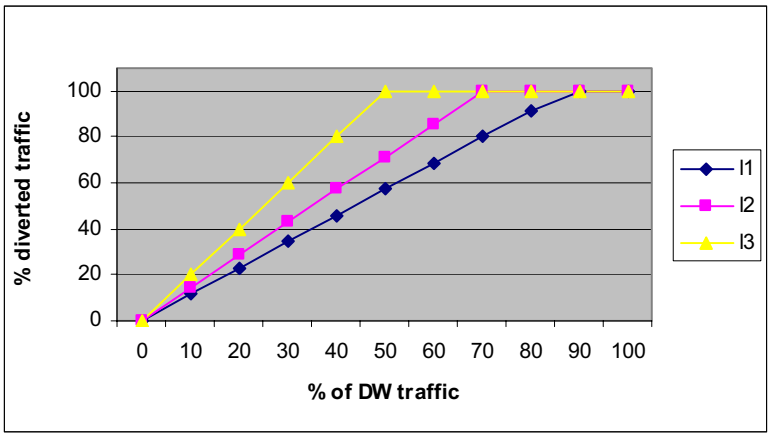


Figure 33. Percentage of diverted P2P traffic depending on amount of P2P_DOWN traffic

6.2 Available bandwidth

To evaluate the gain of the P2P traffic diversion mechanism, the available bandwidth in link *i* (Ai) [47] also referred to as the unutilized capacity, will be calculated for both modes. General expression for available bandwidth:

$$A_i = c_i(1-\rho_i) \qquad (14)$$

Where ρ_i is such that $0 \le \rho_i \le 1$ and it is the load of the link *i* during time interval τ .

6.3 Loss probability

Traffic congestion is the main problem caused by P2P traffic in the links of the aggregation network. Network congestion can be inferred from the occurrence of dropped packets. In queuing theory, the occurrence of dropped packets is called the loss probability. Loss probability can be expressed as:

$$P_{loss} = \frac{\rho^B(1-\rho)}{1-\rho^{(B+1)}} \qquad (15)$$

Where B is the switch buffer size.

6.4 Statistical results

6.4.1 Exit intensity for single switch interaction

The exit intensity was derived for traffic flowing between switches of different switch domains. Tables 12-14 show these interactions. From the tables it can be seen that the *P2P diversion algorithm restricts P2P traffic within a switch domain if the communicating peers are connected to the access nodes belonging to that switch domain.* P2P traffic doesn't create any intensity in the links above the topmost switch of a switch domain and hence conserves network resources that would have otherwise have been used (as with McCircuit mode). In all cases, P2P traffic never flows to the EN

6.4.2 Exit intensity for inter-switch domain interaction

To be able to see the effect of P2P diversion on the scale of the whole topology in a *special case* where users of each switch are exchanging traffic with all the other users of the network, the exit intensity for inter-switch domain interaction was derived.

Table 15 shows the magnitude of P2P traffic intensities for each level of the switch domain. From the table it can be seen that in McCircuit mode the value of the P2P traffic intensity will always be a constant value which is dependant only on the number of access nodes in the switch domain. When the P2P diversion algorithm is enabled in the access nodes, the magnitude of P2P traffic intensities for the respective switch domains becomes a function of the number of access nodes and of the ratio of UP_P2P/DOWN_P2P traffic. DOWN_P2P is the ratio of P2P traffic destined for users belonging to the local switch domain, while UP_P2P is the ratio of P2P traffic flowing to the uplink and is destined to other switch domains. Figure 32 shows that the amount of P2P traffic flowing towards the EN from a switch with the diversion algorithm enabled will be depend on the ratio of upstream/downstream P2P traffic mix and not simply on the number of access nodes in the topology.

6.4.3 Available bandwidth

From the values of the exit intensities in tables 6-8 it can be inferred that for uplinks of the topmost switches of switch domains the available bandwidth is greater when the P2P diversion algorithm is enabled. Using (14) this can be expressed as:

$$A_{ip2p\_mode} > A_{iMcC\_mode}$$

This expression shows that unutilized bandwidth is made available to outgoing traffic due to P2P traffic diversion in the links of the aggregation network.

6.4.4 Congestion

If b, c, and B are assumed constant in expression (15), then λ (traffic intensity) is the main factor contributing to the load in the uplink, and hence to the loss probability. As tables 12-14 show, the P2P traffic intensity will be absent in the uplink of the topmost switches of switch domains if both users are connected to the same switch domain. Hence the loss probability will be lower when P2P diversion is enabled in ANs as compared to the McCircuit mode. With the reduction of load on the links of the access network, the loss probability decreases (15). It can thus be concluded that P2P diversion algorithm reduces congestion in the access network by reducing the traffic load.

6.4.5 Change of network parameters

If there is an increase in the number of users then the amount of traffic intensity generated per AN will increase and hence the traffic load will increase in the respective switches of the aggregation network. An increase in number of users could also lead to a higher probability of users of the network exchanging P2P traffic, this way the probability of intra-switch domain exchange will increase, hence positively affecting available bandwidth and congestion situation.

CHAPTER 7. EXPERIMENTAL DEMONSTRATION

The main aim of the experiment is to illustrate the implementation of the P2P traffic diversion feature in the ELN 200 switch which acts as the access node.

7.1 Overview

Figure 34 illustrates the main components of the experimental set up. The EN sends PAMP commands containing the traffic identities of hosts engaged in P2P traffic exchange to the access node. Using this information, the access node creates and maintains a table of P2P connections. The P2P logic uses the P2P table to modifiy and redirect packets to the respective P2P host peer. This way P2P traffic is made to flow to the directly connected peer and not to the uplink port as McCrcuit logic would otherwise dictate.

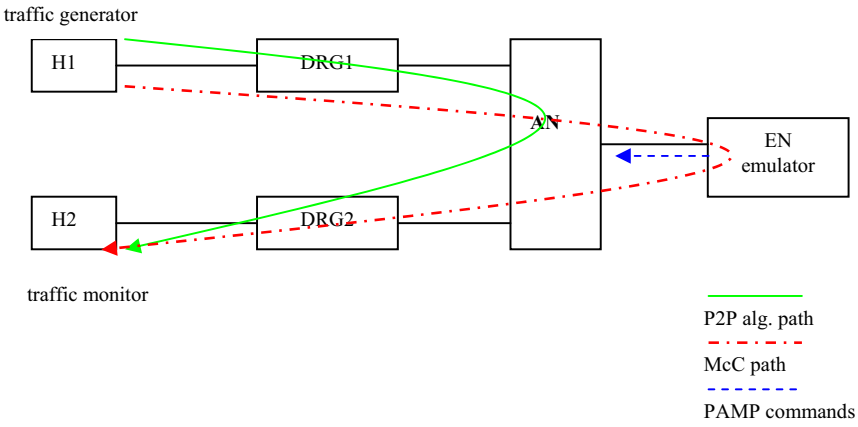


Figure 34. Components of demo set up

The DRGs (from the traffic generator and to the monitor) are configured to send and receive tagged traffic to the access node. Untagged packets flow from the DRG to the host machines. Port 2 of each DRG was set to untag egress packets. While the WAN port of the DRG was set to tagged packets with VLAN tag 8. The configuration settings were: 8;0;0x0a or 8 – VLAN tag, 0 - priority bit, and 0x0a – code for port 2.

7.2 Edge node emulator

In the experimental setup a program written in Erlang emulates the funtionality of the EN. In particular, it provides a GUI to input the values of the hosts' identies. Also using this GUI, two P2P PAMP comands can be sent to the access node. The screen shot of the GUI is presented in figure 35.

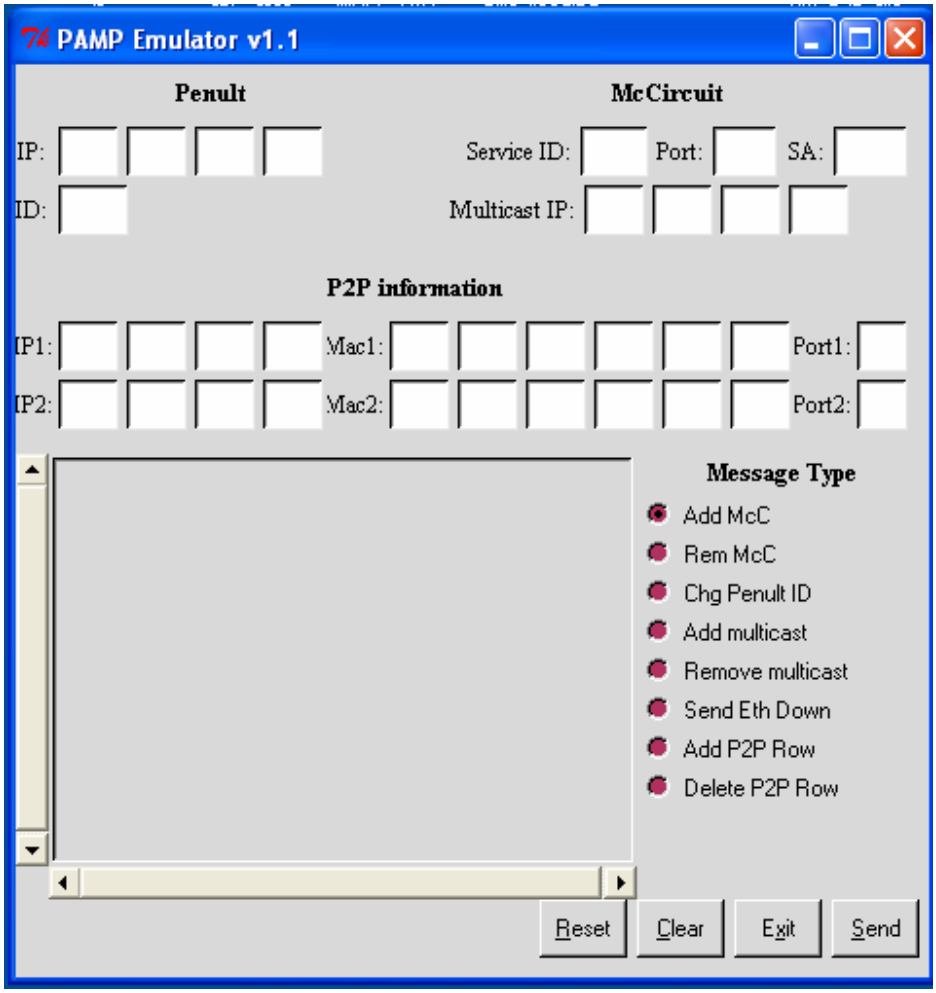


Figure 35. PAMP emulator GUI

7.3 Access node

The access node logic was modified to accommodate P2P logic. This was implemented by creating additional functions and making modifications to the code of the Erlang processor. The main changes were made in the 'pamp_resource.erl' file and an additional function was added to the 'broadcast_resource.erl' to process unicast traffic packets.

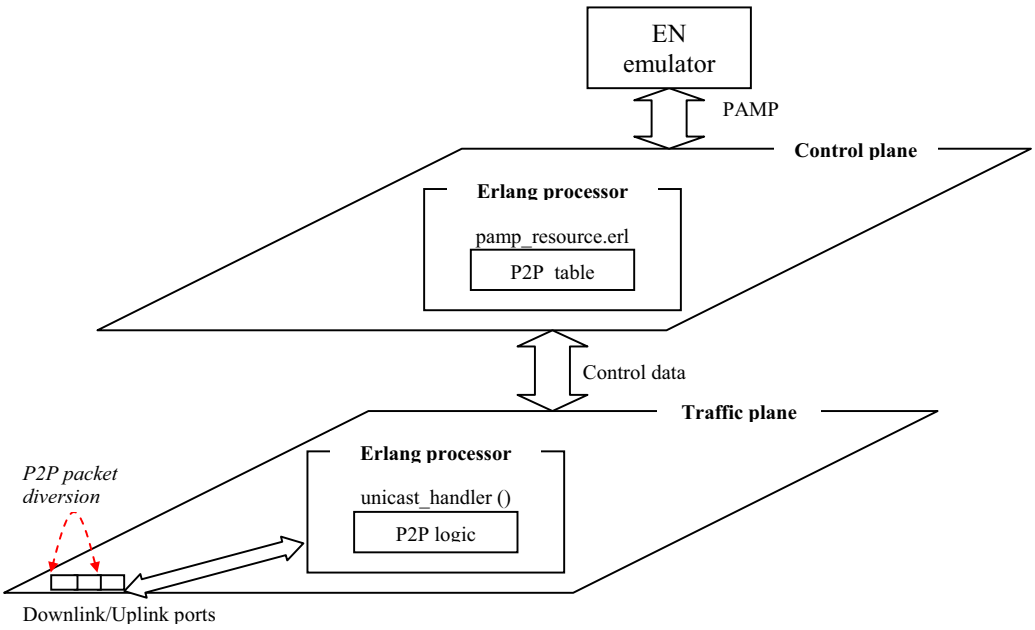


Figure 36. Control and traffic planes of AN with P2P support

The main functions of 'pamp_resource.erl' are to process PAMP messages and invoke the appropriate functions in the access node. Appendix 4 shows this code.

The access node captures all ingress packets, it checks if they match any entry in the P2P table. If there is a match, the destination MAC address is modified according and the packet is switched to the port as specified in the 'd.port' entry of the P2P table. This algorithm is shown in figure 37.

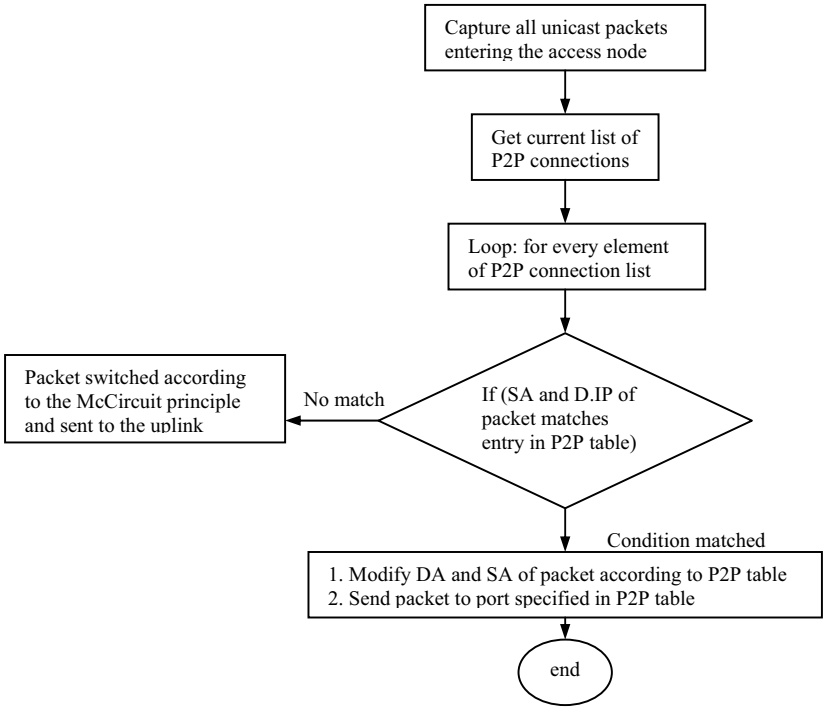


Figure 37. P2P diversion algorithm in McCircuit mode

The logic of the above algorithm is implemented in the unicast_handler function. The listing of this function is in appendix 5.

Due to the fact that the traffic plane in the demo was implemented using the Erlang processor, it was not feasible to override the McCircuit logic as proposed in section 4.2.1.1., as the McCircuit logic is executed in the network processor and this project did not aim do make modifications to the network processor.

7.4 Host traffic emulation

In the experiment host traffic was emulated using a traffic generator called LAN Traffic v.2 [60]. The *trail* version of the software was used. "LanTraffic V2" is a connection and data generation tool for IP networks. The package can emulate TCP or UDP connections between two hosts. Connections can be generated following two different testing modes:
1. Unitary mode: In this mode the traffic generator can be selected. Also packet size and inter-packet delay for each connection can be configured.
Three different data sources are available:

- Automatic data generator by using mathematical formulas,
- Packet generator: different parameters can be defined (number of packets to send, inter-packet delay, packet contents, etc)
- File: selection of a file to send.

2. Automatic mode: In this mode a mathematical formula for connections generation can be selected. The starting time and another mathematical formula for the data volume is specified.

7.5 The Demonstration

In the beginning of the experiment the PAMP emulation GUI was used to populate the P2P table with the traffic identities of the generator and monitor hosts. Ethereal [61], network-monitoring software was started on both host machines. LanTraffic2 software was also started on both hosts. The traffic 'generator' host had LanTraffic2 in send mode, while the 'monitor' host had the software in receive mode. UDP traffic was generated with the destination IP address set to the IP address of the monitor host. Because of the absence of a real edge node, the service binding of each was emulated. The IP address of the default gateway was set to a random IP address. A McCiruit address was generated by the access node and manually entered in the ARP tables of the hosts with that MAC set as the MAC address of the default gateway. This way when the destination IP address is not in the same IP subnet as the host, packets would be forwarded to the default IP address with the corresponding MAC. In so doing the UDP packets generated by the 'generator' host where sent to the access node. At the access node the P2P logic matched the destination IP and source MAC address of incoming packets with information in its P2P table. All packets that matched had their destination MAC addresses modified according to the P2P table. These packets were switched to the port specified in the P2P table. An easily recognized value was used in the P2P tables as the new DA, so that once the packets reached the monitor host, it was easy to see that the destination MAC address was changed and that the packet arrived at the intended port of the access node.

7.5 Experiment results

When traffic from the generator host reached the AN, all packets that matched the previously set of traffic identities were modified according to the P2P table and sent directly to the port to which the monitor host was connected to. At the monitor host it could be seen that the packets were coming from the generator host as they had this destination's MAC address according to the rules in the P2P table. This behavior is a change from the way McCircuit switches packets, i.e. which would have forced them all to flow to the uplink regardless of their destination.

The result of the demonstration was that the P2P algorithm diverts all P2P packets directly to the appropriate downlink, this reduces the number of packets egressing the uplink. Among the benefits of this is that P2P traffic of neighboring hosts is switched directly at the access edge and does not flow through the aggregation network to be switched at the edge node. The consequence is that network congestion that could otherwise have occurred in the links of the aggregation network is reduced.

The theoretical explanation for this is in the fact that congestion is inferred as the occurrence of dropped packets and depends on the rate of incoming packets per unit time, or the traffic intensity of the link. P2PDA reduces the rate of packets egressing the uplink by redirecting them to the downlink and hence reduces the probability of congestion occurring in the uplink.

CONCLUSIONS AND FURTHER WORK

8.1 Conclusions

Traffic generated by P2P applications is responsible for a large percentage of traffic in the aggregation network. This traffic creates contention for best-effort traffic in the access network during peak hours.

While the different methods of P2P control have strong and weak points, a successful P2P solution for broadband networks needs to be a combination of these methods, because each solution by itself doesn't address all aspects of the problem.

Most of the proposed solutions for P2P traffic control (policy management, tiered services, traffic shaping, and blocking) require a mechanism to first identify that the traffic in question is indeed P2P. The growing tendency of traffic encryption by P2P applications makes it impossible to positively identify it. The popularity of P2P applications will continue to grow. Network operators who chose to solve the P2P problem by banning P2P in their networks will not be successful, as most P2P traffic would 'hide' itself and still generate large volumes of traffic.

The P2P diversion algorithm is a solution proposed for identifying and controlling P2P traffic based on the information contained in layer 2 and layer 3. Other policies could be easily added to narrow the definition of P2P traffic at the edge node.

Analytical results of the P2P diversion algorithm for networks employing McCircuit show that congestion produced by P2P traffic in certain links of the aggregation network could be significantly reduced provided that the peers are connected to the access network.

The positive effect of the P2P diversion algorithm is manifested when there is intra-switch domain traffic exchange of P2P traffic. In the worst-case scenario, the P2P traffic diversion algorithm can prevent P2P traffic from flowing to the EN. In the best-case scenario, P2P traffic can be restricted to the switch domain in the lowest level of the aggregation switch hierarchy. The shorter the distance between end switches (smaller the switch domain) to which the users are connected to, the more the P2P traffic is localized.

The P2P diversion algorithm brings a finer tuning capability to traffic flow control in the aggregation network where traffic separation based on McCircuit is used. By enabling the P2P diversion algorithm in the access node, the network operators can control the amount and type of traffic at the edge node. The cost of service per user for the network operators will be reduced due to the fact that increased traffic volumes generated by users would not lead to the need to expand capacities at the EN since P2P traffic never reaches it.

From a business point of view, the available bandwidth created by P2P diversion could be used by access network providers to increase revenues from their existing infrastructure – by offering new services, attracting new customers (both end users and service providers), and retaining existing customers.

P2P diversion could greatly reduce Internet peering costs that are currently a major cost for Internet service providers due to the growth of P2P traffic. The algorithm diverts all P2P traffic locally and a relevant pricing scheme will encourage users to share content within the access network as much as possible.

The proposed business model retains the flat rate pricing scheme popular among broadband users and allows the network operator to gain revenue from P2P traffic.

The proposed algorithm works in an IPv4 environment. In a scenario where IPv6 addressing is used, there will be no need for the P2PDA algorithm. P2P packets could be forwarded in the access network by the aggregation network switches based on the 64-bit interface identifier of the IPv6 address [66].

8.2 Further work

1. One of the major investigations that is left undone is to implement the P2P diversion algorithm in the network processor of the access node. This task would require additional coding in assembler language and testing afterwards.
2. Enhance the PAMP to support message initiation by the access node towards the edge node. It has been observed that there are certain events that occur at the access node that could be useful for the edge node to be aware of. An example is when a P2P table entry expires. Also sending information such as the byte count at a certain port of the access node could be of use.
3. Solution to P2P traffic control in IPv6 environment. With the growing tendency of shifting to IPv6, there is need to investigate how to deal with P2P in an IPv6 network. In particular, it has to be investigated how the network operator would implement accounting for P2P traffic since it wouldn't have to go via the edge node where the account functionality is currently implemented.

REFERENCES

[1] J. A. Clark and A. Tsiaparas, “Bandwidth-on-demand networks — a solution to peer-to-peer file sharing”, BT Technology Journal archive Volume 20, Issue 1, January 2002

[2] C. Soldani, “Peer-to-Peer Behavior Detection by TCP Flows Analysis”, Engineer dissertation, University of Liege, France. 2004

[3] D. S. Milojicic, V. Kalogeraki, R. Lukose, K. Nagaraja, J. Pruyne, B. Richard, S. Rollins, and Z. Xu., “Peer-to-peer computing.” Technical Report HPL-2002-57, HP Lab, 2002.

[4] B. Yang and H. Garcia-Molina, “Comparing Hybrid Peer-to-Peer Systems”, The VLDB Journal, September 2001, pp 561-570.

[5] Internet2 Netflow, Weekly Reports, Weeks of 20020218, 20020923, 20030505 http://netflow.internet2.edu/weekly, May 2003

[6] A. Gerber, J. Houle, H. Nguyen, M. Roughan, and S. Sen, “P2P: the gorilla in the cable,” in Proc. National Cable & Telecommunications Association (NCTA), June 2003

[7] S. Sen and J. Wang, “Analyzing peer-to-peer traffic across large networks,” in Proc. Second Annual ACM Internet Measurement Workshop, Nov. 2002, pp. 137–150.

[8] KaZaA supernodes: www.realityradio.org/ftfakes/kazupernodes/ Accessed 15.10.2004

[9] R. Schollmeier and A. Dumanois, “Peer-to-Peer Traffic Characteristics” , Lehrstuhl für Kommunikationsnetze, Technische Universität München. Germany, 2003

[10] C. Shirky, “What is P2P... and what Isn’t. An article published on O’Reilly Network. www.openp2p.com/lpt/a//p2p/2000/11/24/shirky1-whatisp2p.html. Accessed 11.10.2004

[11] Marc Morin, “Managing P2P Traffic on DOCSISTM Networks”, Sandvine Incorporated. February 2002.

[12] CAIDA home page, http://www.caida.org, Accessed 11.10.2004

[13] Sandvine Incorporated ,"Peer-to-Peer File Sharing The impact of file sharing on service provider networks", An industry whitepaper, Ontario, Canada, December 2002

[14] Sandvine Incorporated, “Meeting the Challenge of Today’s Evasive P2P Traffic”, An industry White Paper, Ontario, Canada, September 2004

[15] A. Abimbola, Q. Shi, and M. Merabti, Using Intrusion Detection to Detect Malicious Peer-to-Peer Network Traffic, in PGNET 2003, Manchester, UK, June 2003.

[16] P-Cube, Approaches to Controlling Peer-to-Peer Traffic: a Technical Analysis, technical report, 2003. http://www.p-cube.com. Accessed 11.10.2004

[17] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Schenker, “A Scalable Content-Addressable Network”, in Proceedings of the 2001 conference on applications, technologies, architectures, and protocols for computer communications, ACM Press, 2001, pp. 161-172.

[18] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan, “Chord: A Scalable Peer-to-Peer Lookup Service for Internet Applications”, in Proceedings of ACM SIGCOMM, ACM Press, 2001, pp. 149/160.

[19] Sandvine Incorporated, “Regional characteristics of P2P: File sharing as a multi-application, multi-national”, technical report, October 2003. http://www.sandvine.co.uk/solutions/download_center.asp. Accessed 11.10.2004

[20] I. Clarke, O. Sandberg, B. Wiley, and T. W. Hong, “Freenet: A Distributed Anonymous Information Storage and Retrieval System”, Lecture Notes in Computer Science, 2001, p.46.

[21] SETI@HOME 2001. setiathome.ssl.berkeley.edu. Accessed 11.10.20

[22] NAPSTER. 2001. The Napster home page, www.napster.com. Accessed 11.10.2004

[23] GNUTELLA. 2001. The Gnutella home page, www.gnutella.com. Accessed 11.10.2004

[25] D. Strom, “Businesses Embrace Instant Messaging”. January 2001

[26] Sprint. Packet Trace Analysis. http://ipmon.sprintlabs.com/. Access 22.11.2004

[27] N. Manjaro, "Peer to Peer networking", working text for Architecture & Transport Working Group, DSL Forum, August 2004.

[28] Digital Intelligence Centre, news archive. http://www.itic.ca/DIC/News/archive.html. Accessed 22.11.2004

[29] MUSE work document, “McCircuit details”, IST – 6th FP, October 2004

[30] S. Waterhouse,D.M. Doolin,G. Kan,Y Faybishenko, "Distributed Search in P2P Networks", IEEE Internet Computing 6(1):68-72. January-February 2002.

[31] BUZZPAD, Buzzpad home page, www.buzzpad.com. Accessed 25.11.2004

[32] SKYPE, Skype homepage, www.skype.com. Accessed 25.11.2004

[33] Quazal, Quazal homepage, www.quazal.com. Accessed 25.11.2004

[34] PlanetDecent, PlanetDecent homepage, www.planetdescent.com. Accessed 25.11.04

[35] Cybiko, Cybiko homepage, www.cybiko.com. Accessed 25.11.2004

[37] EDONKEY, eDonkey homepage, www.edonkey.com. Accessed 25.11.2004

[38] FREENET, Freenet website, http://freenet.sourceforge.net/ . Accessed 25.11.2004

[39] IMESH, iMesh website, www.imesh .com. Accessed 25.11.2004

[40] KAZAA, KaZaa website, www.kazaa.com. Accessed 25.11.2004

[41] Fastrack, http://en.wikipedia.org/wiki/FastTrack, Accessed 25.11.2004

[42] A. Oram, “OpenCola: Swarming Folders”, www.openp2p.com/pub/a/p2p/2001/05/24/oram.html, O’Reilly networks, Accessed 25.11.2004

[43] J. Teig von Hoffman, "Guide to Distributed PowerPoint", Access grid documentation project, Boston University, July 2003

[44] Scour Exchange, Scour exchange website, www.scour.com. Accessed 25.11.2004

[45] B. Y. Zhao, Y. Duan, L. Huang, A. Joseph, and J. Kubiatowicz,"Brocade: Landmark Routing on Overlay Networks," in Proceedings of the First International Workshop on Peer-to-Peer Systems (IPTPS 2002). 2002.

[46] X. Zhichen, M. Mahalingam, and M. Karlsson, "Turning Heterogeneity into an Advantage in Overlay Routing," Proc. INFOCOM. 2003.

[47] P. B. Keating, G. G. Cappiello, K. J. McIntyre and Y. A. Yudin, "Feeding fiber-starved networks", Confluent Photonics Corp, in CED magazine, June 2004.

[48] C. Dovrolis P. Ramanathanm and D. Moore, “What do packet dispersion techniques measure?”, in Proceedings of IEEE INFOCOM, 2001.

[49] Network traffic statistics, Phantom project, Ericsson internal report 410, 2003

[50] Bredbandsbolaget SLA, http://www.bredband.com/se/content.jsp?t=2&s=3&m=302, Accessed 04.04.2005

[51] Direct Connet, DC++ website, www.dcplusplus.com, Accessed 05.04.2005

[52] Muse Project Deliverable DA 1.1, "Towards multi-service business models", IST - 6th FP, June 2004

[53] R. Korfhage. Information Storage and Retrieval. John Wiley, 1997.

[54] M. T. Schlosser, T. E. Condie, and S. D. Kamvar, "Simulating a File-Sharing P2P Network", Stanford university, First Workshop on Semantics in P2P and Grid Computing, December, 2002

[55] L. Breslau, P. Cao, G. Phillips, and S. Shenker, Web caching and Zipf-like distributions: and implications,” in INFOCOM 1999.

[56] Slyck, P2P application statistics, http://www.slyck.com/stats.php, Access 06.04.2005

[57] EuroTelcoblog, "Skype tipping point?", 07.04.2005 http://eurotelcoblog.blogspot.com/2005/04/skype-tipping-point-yep-richard-asked.html Accessed 08.04.2005

[58] MUSE project deliverable DA 2.2, “Network architecture and functional specifications for the multi-service access and edge”, IST - IST - 6th FP, January 2005

[59] K. Keys, D. Moore, R. Koga, E. Lagache, M. Tesch, and K. Claffy. “The architecture of the CoralReef: Internet Traffic monitoring software suite”, PAM, 2001

[60] Ethereal, Ethereal website, www.etherael.com. Accessed 17.04.2005

[61] LANTrafficV2, Omnicor website, www.omnicor.com/netest.htm, Accessed 17.04.2005

[62] C. Courcoubetis and V. A. Siris, "Managing and Pricing Service-Level Agreements for Differentiated Services," 7th Int'l Wkshp. on QoS (IWQoS'99), 1999

[63] H. Sigurdsson, "Service differentiation in residential broadband networks", draft, CTI, DTH. October 2004

[64] P-Cube Engage: Tiered Service Control, P-Cube website, www.p-cube.com. Accessed 17.04.2005

[65] K. Sripanidkulchai, "The Popularity of Gnutella Queries and Its Implications on Scalability". White paper, O'Reilly website, February 2001. www.openp2p.com/topics/p2p/gnutella Accessed 20.04.2005

[66] G. Q. Maguire Jr., " Personal Computing and Communication: It is more than just networking of mobile devices", Networking 2002: The Second IFIP-TC6 Networking Conference, Pisa, Italy, May 24 2002

APPENDIX 1. Algorithm: hosts connected to one AN

User 1	User 2	AN	EN
MAC1 IP1 DGW=IPen ARP table: IPen ->McC1 AN port 1	MAC2 IP2 DGW=IPen ARP table: IPen ->McC2 AN port2		User1->McC1 – IP1 User2->McC2 – IP2
1.User1 wants to send P2P traffic to user2 ARPREQUEST: SA=MAC1 DA=McC1 S.IP=IP1 D.IP=IP2			
		Send to respective SA of EN	
			1.If (S.IP&D.IP ∈ same subnet) {create table: possible_P2P_flow SA |D.IP|DA MAC1|IP2 |MAC2 MAC2|IP1 |MAC1 (mirror) } 2. Send all traffic to user2 via EN SA=McC2, DA=MAC2 3. Create P2P table in AN using PAMP ADDP2PRW 4. If user1 or user2 terminates service binding, send PAMP REMP2PRW
	Gets traffic from user1 ARP table: IP1->McC2		

		P2P_table updated in RT:
		Check | Substitute
		SA |D.IP| s.port | DA | SA
		MAC1|IP2 | 1 | MAC2 |McC2
		MAC2|IP1 | 2 | MAC1 |McC1
		Filter incoming packets:
		If (SA&S.IP&D.IP∈ a flow)
		{change:
		SA-> SA_t (McC2)
		DA-> DA_t (MAC2) }
		
		If(no traffic user1<->user2 after T_age)
		{remove entries in the P2P_table}	

APPENDIX 2. Algorithm: hosts connected to two ANs

User 1	AN1	User2	AN2	EN
MAC1 IP1 DGW=IPen ARP table: IPen->McC1 AN 1, port 1		MAC2 IP2 DGW=IPen ARP table: IPen ->McC2 AN2, port1		User1->McC1 – IP1 User2->McC2 – IP2 DB of all IP @ of CN users
1.User1 wants to send P2P traffic to user2 ARPREQUEST: SA=MAC1 DA=McC1 S.IP=IP1 D.IP=IP2				
	Send to respective SA of EN			
				1.If (S.IP&D.IP ∈ belong to pool of IPs) {create tables: out: SA |D.IP |DA MAC1 |IP2 |MAC2 in: SA |D.IP |DA MAC2 |IP1 |McC1 out: SA |D.IP |DA MAC2 |IP1 |MAC1 in: SA |D.IP |DA MAC1 |IP2 |McC2 } 2.Send all traffic to user2 via EN SA=McC2, DA=MAC2 3. Create P2P tables in ANs using PAMP ADDP2PRW
		Gets traffic from user1 ARP table: IP1->McC2		4. If user1 or user2 terminates service binding, send PAMP REMP2PRW to ANs
	out: SA |D.IP| sp |DA MAC1 |IP2 | 1 |MAC2 in: SA |D.IP |SA | dp MAC2 |IP1 |McC1 | 1 Filter incoming and out going packets according to tables. If(no traffic user after T_age) {remove entries in the P2P_table}		out: SA |D.IP| sp |DA MAC2 |IP1 |1 |MAC1 in: SA |D.IP |SA | dp MAC1 |IP2 |McC2 | 2 Filter incoming and out going packets according to tables. If(no traffic user after T_age) {remove entries in the P2P_table}	

APPENDIX 3. 'pamp_resource.erl'

```
-module(pamp_resource).

%Edited by Ayodele Damola
%Edited by Jonathan Olsson 050114
%Changed so that PAMP traffic is sent to the same port as the source port

%%% This resource handles the PAMP protocol for McCircuit
%%%---------------- PAMPv1 Definitions
%% Penult Apex Protocol Types
-define(PAMP_ACK,       10).
-define(PAMP_NACK,      11).
-define(PAMP_SETID,      3).
-define(PAMP_SNDIP,    102).
-define(PAMP_SNDETHU,  104).
-define(PAMP_SNDETHD,  105).
-define(PAMP_SNDETHR,  106).
-define(PAMP_ADDMC,    110).
-define(PAMP_REMMC,    111).
-define(PAMP_JOIN,     114).
-define(PAMP_LEAVE,    115).
-define(PAMP_SNDTOPORT, 200).
-define(PAMP_SNDBACK,   201).

%% P2P PAMP commands
-define(PAMP_ADDP2PRW, 300).
-define(PAMP_DELP2PRW, 301).

%%%%%%%%
start(State) ->
    process_flag(trap_exit,true),
    InitalState0 = State#state{status = [{last_resp,undefined} | State#state.status],
                               parameters = [{last_seq_no,-1},
                                             {enabled,false},
                                             {peer_ip,undefined} | State#state.parameters]
                              },
    InitalState = case ?START_PAMP_ENABLED of
                      true -> open_pamp_socket(InitalState0);
                      false-> InitalState0
                  end,
    InitalState#state.parent_pid ! {start_ok, self()},
    loop(InitalState, socket(InitalState)).
%%%%%%%%
%% Main loop
loop(State, Socket) when record(State,state) ->
    receive
        UDP = {udp,Socket,_,_,_} ->
            ?MODULE:loop( handle_upd_packet(UDP,State), Socket );

        {set,Ref,From,Params} ->
            case handle_set(Params,State) of
                {ok,New_state} when record(New_state,state)->
                    From ! {reply,Ref,ok},
                    ?MODULE:loop(New_state, socket(New_state));
                Reply ->
                    From ! {reply,Ref,Reply},
                    ?MODULE:loop(State, Socket)
            end;
        {check,Ref,From,Params} ->
            From ! {reply,Ref,handle_check(Params,State)},
            ?MODULE:loop(State, Socket);

        Signal ->
            case resource_system_lib:handle_generic_signal(Signal,State) of
                ok -> ?MODULE:loop(State, Socket);
                {ok,New_state} -> ?MODULE:loop(New_state, Socket);
                {error,Reason} -> ?log_error(Reason,[Signal]), ?MODULE:loop(State, Socket)
            end

    end.
```

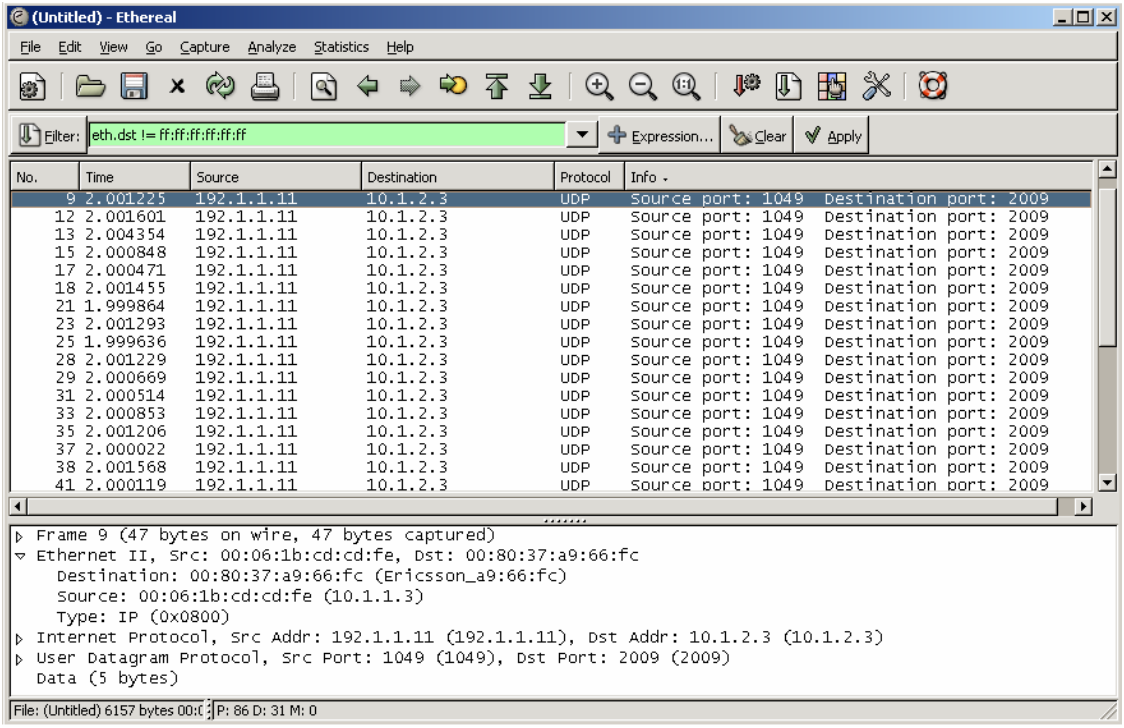
```
%%%%%%%%%%%%%%%%%%%%%%%%% FUNCTIONS %%%%%%%%%%%%%%%%%%%%%%%%%
.
.
.
%%%================================================================
%% Handler of incoming PAMP commands
handle_req_commands(Data) ->
    handle_req_commands(Data, 1).

handle_req_commands(<<Type:16, Len:16, More/binary>>, N) ->
    <<Data:Len/binary, Tail/binary>> = More,
   % io:format("data: ~n~w~n",[Data]),
    case catch handle_req_command(Type, Data) of
        ok ->
            handle_req_commands(Tail, N+1);
        {nok,Reason} ->
            io:format("Failed on nok"),
            ?log_error("PAMP command failed!",{Reason,Type,Data}),
            {nok,N};
        Reason ->
            io:format("PAMP command failed: ~w",[Reason]),
            ?log_error("PAMP command failed!",{Reason,Type,Data}),
            {nok,N}
    end;
handle_req_commands(<<>>, N) ->
    ok.

.
.
.

%% PAMP ADDP2PRW handler
handle_req_command(?PAMP_ADDP2PRW, Data) ->

<<P1mc1:8/integer,P1mc2:8/integer,P1mc3:8/integer,P1mc4:8/integer,P1mc5:8/integer,P1mc6:8
/integer,P1ip1:8/integer,P1ip2:8/integer,P1ip3:8/integer,P1ip4:8/integer,P2mc1:8/integer,
P2mc2:8/integer,P2mc3:8/integer,P2mc4:8/integer,P2mc5:8/integer,P2mc6:8/integer,P2ip1:8/i
nteger,P2ip2:8/integer,P2ip3:8/integer,P2ip4:8/integer>> = Data,

    Mc1 = <<P1mc1,P1mc2,P1mc3,P1mc4,P1mc5,P1mc6>>,
    Ip1 = <<P1ip1,P1ip2,P1ip3,P1ip4>>,
    Mc2 = <<P2mc1,P2mc2,P2mc3,P2mc4,P2mc5,P2mc6>>,
    Ip2 = <<P2ip1,P2ip2,P2ip3,P2ip4>>,

    {ok, TR}=timer:send_after(timer:seconds(20), self(), {timout,{Mc1,Ip1,Mc2,Ip2}}),
    Con = {Mc1,Ip1,Mc2,Ip2,TR},
    %io:format("new p2p entry: ~w ~n",[Con]),

    {ok,Cons} = api:get(mcc_data, con_list),
    if
       length(Cons) =< 0 ->
            api:set(mcc_data, con_list, [Con]),
            io:format("single con added ~n");
       length(Cons) > 0 ->
            api:set(mcc_data, con_list, lists:append(Cons,[Con])),
            io:format("another con added ~n")
    end,

    {ok,Conss} = api:get(mcc_data, con_list),
   % io:format("the data as in ptp_cons: ~n~w~n ",[Conss]);
    io:format("-----------------------------------The P2P table--------------------------
----- ~n"),
    prntl(Conss),
    io:format("-----------------------------------------------------------------------------
---- ~n");

%% PAMP ADDP2PRW handler
handle_req_command(?PAMP_DELP2PRW, Data) ->
    io:format("inside DEL p2p handle_req_command ~n"),
    io:format("the data: ~n~w~n ",[Data]),
```

```
<<P1mc1:8/integer,P1mc2:8/integer,P1mc3:8/integer,P1mc4:8/integer,P1mc5:8/integer,P1mc6:8
/integer,P1ip1:8/integer,P1ip2:8/integer,P1ip3:8/integer,P1ip4:8/integer,P2mc1:8/integer,
P2mc2:8/integer,P2mc3:8/integer,P2mc4:8/integer,P2mc5:8/integer,P2mc6:8/integer,P2ip1:8/i
nteger,P2ip2:8/integer,P2ip3:8/integer,P2ip4:8/integer>> = Data,

    Mc1 = <<P1mc1,P1mc2,P1mc3,P1mc4,P1mc5,P1mc6>>,
    Ip1 = <<P1ip1,P1ip2,P1ip3,P1ip4>>,
    Mc2 = <<P2mc1,P2mc2,P2mc3,P2mc4,P2mc5,P2mc6>>,
    Ip2 = <<P2ip1,P2ip2,P2ip3,P2ip4>>,

    Con = {Mc1,Ip1,Mc2,Ip2},
    %io:format("new p2p entry: ~w ~n",[Con]),

    {ok,Cons} = api:get(mcc_data, con_list),
    C = remove_en(Cons,Con,Cons),
    api:set(mcc_data, con_list, [C]),
    %api:set(mcc_data, con_list, lists:delete(Con,Cons)),
    {ok,Conss} = api:get(mcc_data, con_list),
    %io:format("the data after DELP2PRW in ptp_cons: ~n~w~n ",[Conss]);
    io:format("-----------------------------------The P2P table------------------- ~n"),
    prntl(Conss),
    io:format("--------------------------------------------------------------------- ~n");
handle_req_command(Type, Data) ->
    {nok,unknown_pamp_command}.

find_group(Group,[]) ->
    io:format("Group undefined~n"),
    undefined;

find_group(Group, [{MultiCastGroup, {ok,Data}} | T]) ->
    %io:format("~w match ~w?",[Group,Data]),
    case main_lib:value_from_tuple_list(group,Data) of
        Group ->
            io:format("yes~n~n"),
            {ok,MultiCastGroup};
        _ ->
            %io:format("no~n"),
            find_group(Group, T)
    end;
find_group(_, _) ->
    io:format("Group undefined~n"),
    undefined.

%% Function printing out the P2P table of AN
prntl([X|R])->
     %io:format("H1->H2 ~n"),
     io:format('--- H1->H2 --- \n| SA: ~w| D.IP: ~w| new DA: ~w| \n',
[element(1,X),element(4,X),element(3,X)]),
     %io:format("H2->H1 ~n"),
     io:format('--- H2->H1 --- \n| SA: ~w| D.IP: ~w| new DA: ~w| \n',
[element(3,X),element(2,X),element(1,X)]),
     prntl(R);
prntl([]) -> io:format(" ").
%% Function removing an entry from the P2P table
remove_en([],En,Cons)->
    Cons;

remove_en([X|R],En,Cons)->
    X1={element(1,X),element(2,X),element(3,X),element(4,X)},
    En1={element(1,En),element(2,En),element(3,En),element(4,En)},
    if
     X1 == En1 ->
            io:format("inside abot to del ~n "),
            io:format("element to del ~w ~n",[X]),
            Conss=lists:delete(X,Cons); %save the edited con_list in mcc_data no need to
return Conss
      true->
             remove_en(R,En,Cons)
     end.
```

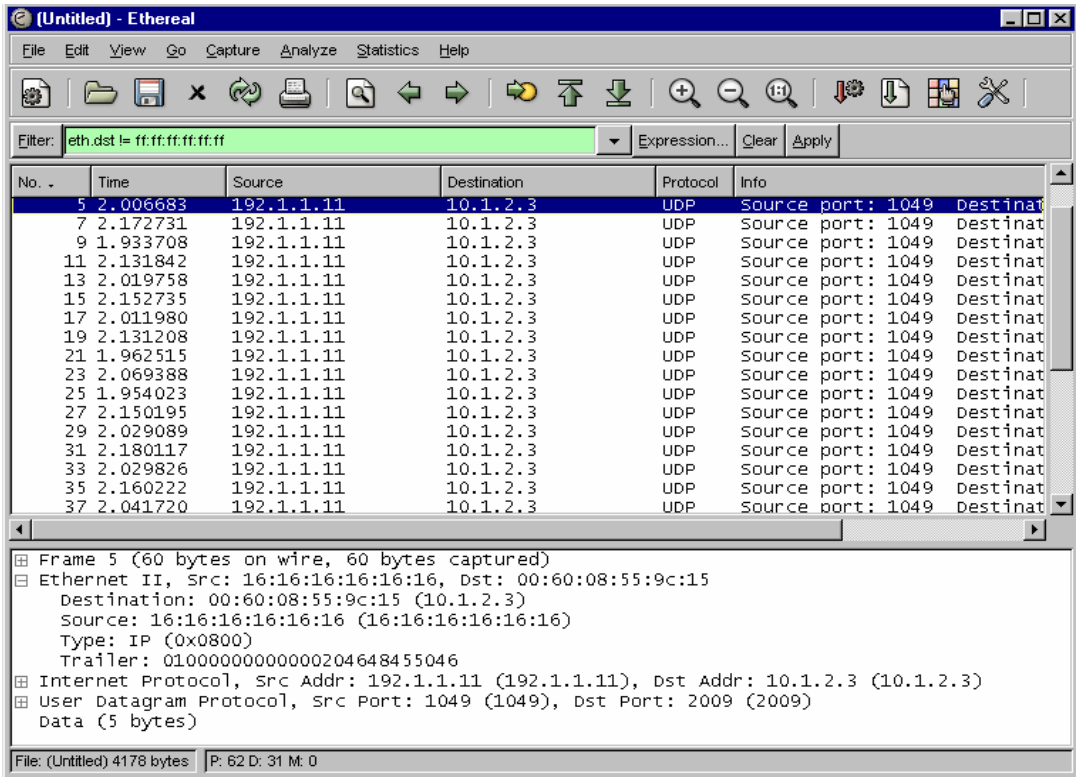
APPENDIX 4. ‘unicast_handler()’

```
%% Function modifies ingress packets if it matches a profile in the P2P table
unicast_handler(Packet,Tag,Ptag) ->

    io:format(" SA: ~w ~n DA: ~w ~n S_IP: ~w ~n D_IP: ~w ~n ~n",[SA,DA,S_IP,D_IP]),

    Cons = case api:get(mcc_data, con_list) of
              {ok, Values} -> Values;
              _ -> []
          end,
    %io:format("cons: ~w ~n",[Cons]),

    if
       length(Cons) > 0 ->
           case get_new_da(Cons,SA,D_IP) of
                       {ok,Pa} ->

                            io:format("got P2P packet, Modifing DA.... ~n"),
                               %% Modifiying packet
                            Newpacket = <<Pa:6/binary,Rst/binary>>,
                               %% Sending to port AN to which the other peer is connected
to
                            P={ethernet_port,user_port},
                              api:call(P, send_packet, Newpacket, 8, Ptag),
                            io:format("sent P2P packet to port user_port ~n");
                       false ->

                            true
                   end;
       length(Cons) =< 0 ->
           false
    end.

%%--------------------------------------------------------------------

%% Functions extracts new destination MAC address from the P2P table
  get_new_da([],MC,IP) ->
       % io:format("no match ~n"),
        false;

    get_new_da([X|R],MC,IP)->
      %  io:format("in new_da fx. SA: ~w DIP: ~w ~n",[MC,IP]),
       % io:format("elm1: ~w elm4: ~w ~n",[element(1,X),element(4,X)]),
       % io:format("elm3: ~w elm2: ~w ~n",[element(3,X),element(2,X)]),
       if
              MC == element(1,X), IP == element(4,X)->
               %      io:format("match between 1 and 4 conlist ~n"),
                     {ok,element(3,X)};                          %newDA=Mac2

              MC == element(3,X), IP == element(2,X)->
              %      io:format("match between 3 and 2 conlist ~n"),
                      {ok,element(1,X)};                          %newDA=Mac1
       true->
              get_new_da(R,MC,IP)
       end.
```

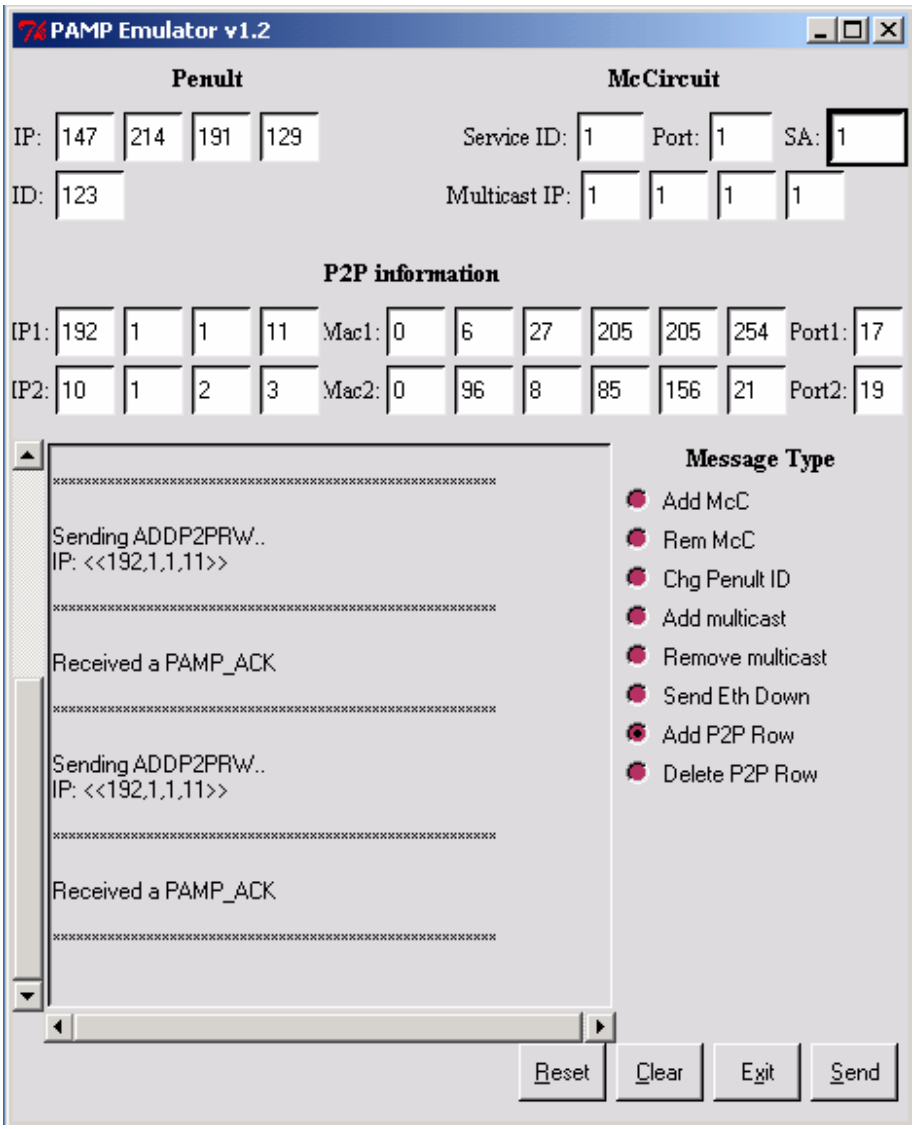
APPENDIX 5. Screen shot of traffic sending host



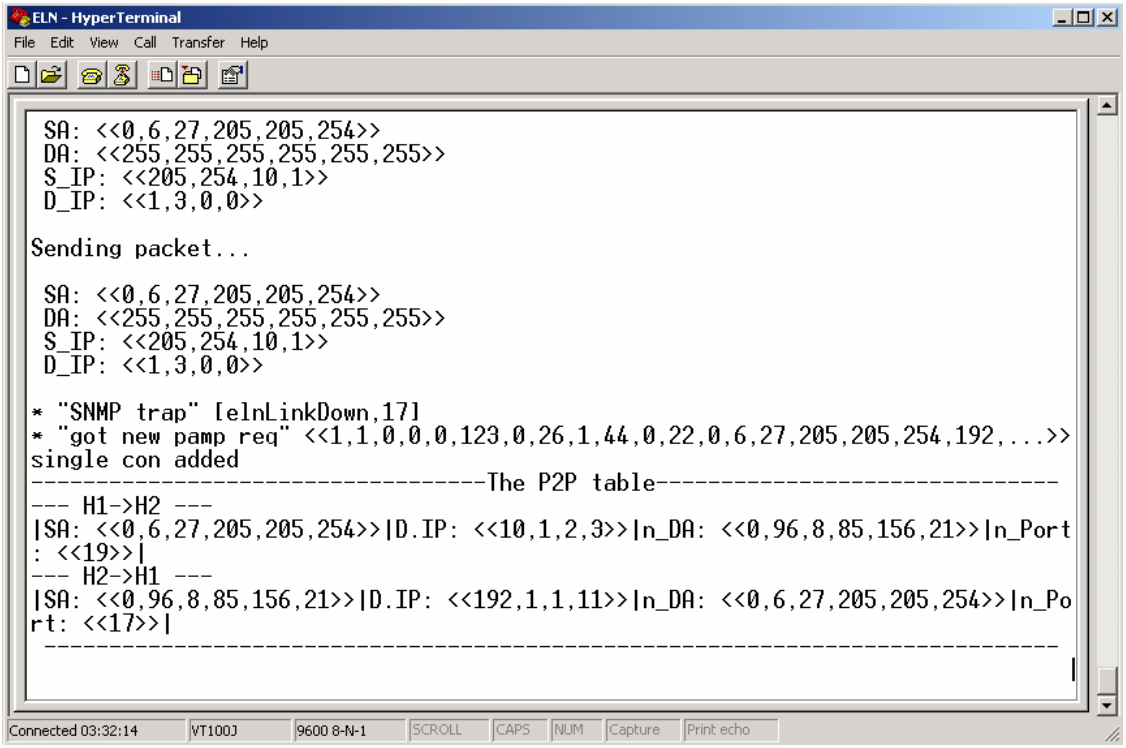
APPENDIX 6. Screen shot of traffic receiving host



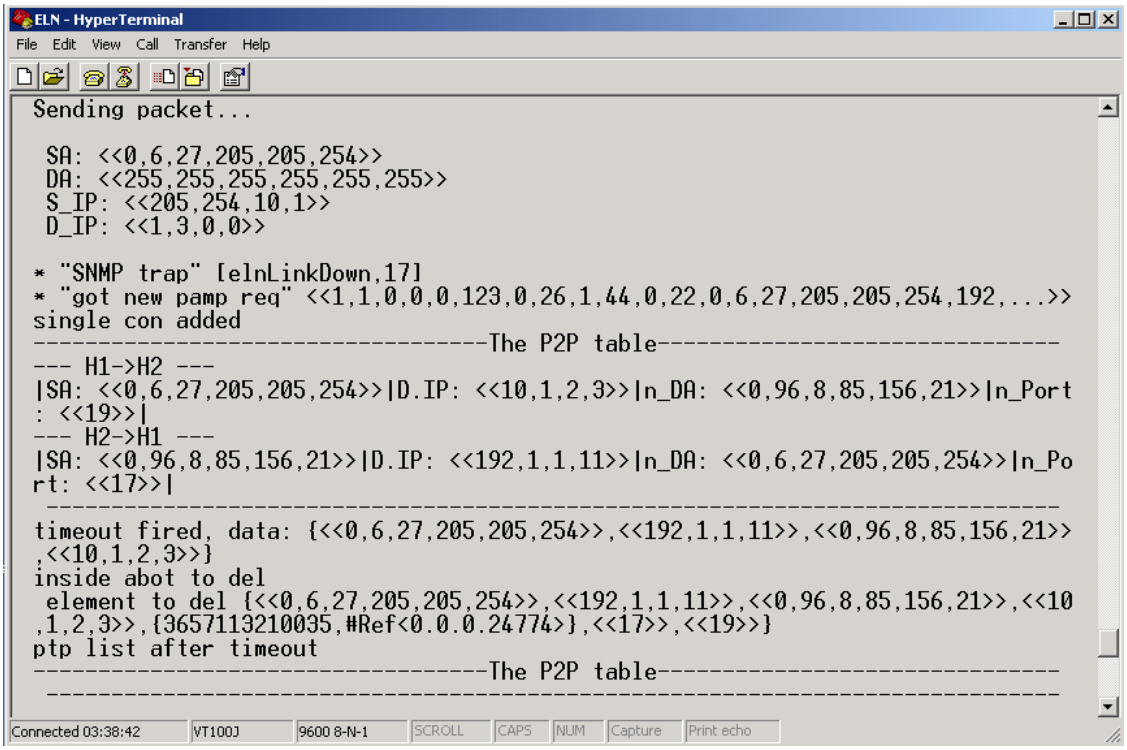
APPENDIX 7. Screen shot of EN emulation



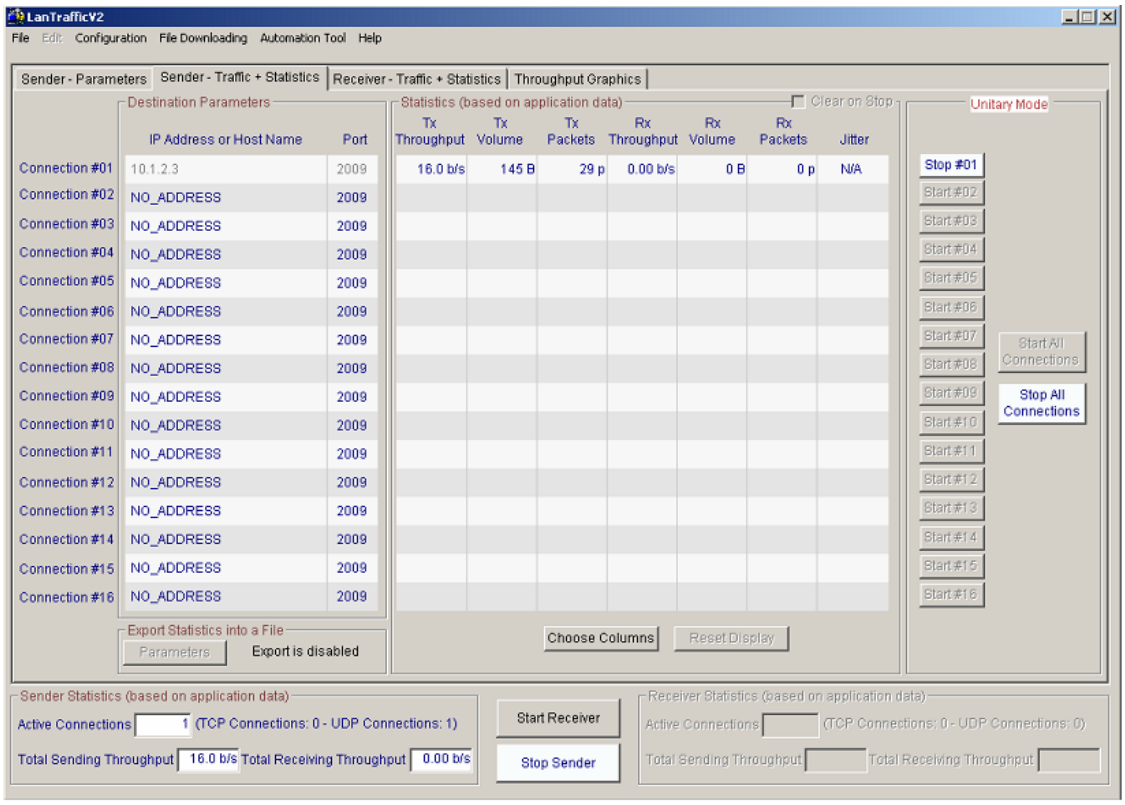
APPENIX 8. Screen shot of remote connection to AN, new entry in P2P table



APPENDIX 9. Screen shot of remote connection to AN, P2P entry timeouted out



APPENDIX 10. Screen shot of UDP traffic generator software



IMIT/LCN 2005-10

www.kth.se