Efficiency Issues in a Switched LAN

Muhammad Mustafa Hassan
mhassan@abo.fi

Luigia Petre
lpetre@abo.fi

Department of Information Technologies, Åbo Akademi University

***Abstract*—Ethernet—a switched local area network—is the market leader for LAN Technologies today. From businesses to government organizations, home to shopping centers, and NGOs to universities, Ethernet is the sole shareholder of the local area networks. Due to such a wide-range, Ethernet has become an intensively addressed research area, with concerns ranging from the high level network design problems to the mid-level theoretical protocols architectures to the low level physical hardware details. As a result, new standards, increased speeds and better technology for Ethernet are very often reported.**

Due to the fast and intensive development, the diameter of the LAN deployment has been increased to a bigger scale. Local area networks have grown from a couple of hundreds of clients to several thousands of clients advancing the concept of a Campus Area Network. The huge enhancements in network size—and constant growth—led to various concerns in the switched network design, some very crucial. First, the fault tolerance and the availability of the network should be ensured for virtually 100% of the time. Second, the aggregation of the switched traffic flow to the network core and the Internet is a major aspect, as the proper size of upstream links keeps the network from being congested. Third, the proper design of the switched network has quality, speed and efficiency aspects to consider.

The above mentioned issues do not only arise when a network is being deployed. They can also arise after the establishment of a network, due to several reasons like arbitrary growth, careless planning, design mistakes, replacement of devices/links upon failures or for extending, etc. All these concerns lead to one common theme: that of the efficiency of the LAN.

In this paper, we provide a case study of a university LAN: we set out to uncover just how efficient the Åbo Akademi University's Ethernet is. The results we present are obviously characteristic to this particular network. We found that the network design of the Åbo Akademi LAN was influenced over the years by the limited budget as well as the lack of stringent security and efficiency needs. While there is no reason to generalize our findings, we can still draw several lessons from our study. We have observed various efficiency problems and proposed solutions to them. By simulation we have also visually illustrated the differences and improvements our solutions would bring.

Index Terms—Ethernet, hierarchal design, LAN, Switched LAN, Switching, ÅA Network

I. Introduction

Computing and communication have had an essential role in people's lives from the early ages. Starting from abacus and tally marks to the room-sized ENIAC, and then to the PC and dust computing devices, a drastic revolution took place in the field of computing. The field of data communication also transformed radically—from lighted beacons, smoke signals and homing pigeons to telegraph and radio, and then to the computer networks and the Internet. During the early stages, the means for storage, processing and communication of data were distant. Humans used to store data in their minds, and later on papers. For computing they needed some abacus type of machines, while for communications they needed drums or smoke signals and later the telegraph. Later still, with the advent of the computers and the stored program concept, the means of storage and processing merged. Still, the communication was isolated. The forms of data to be stored and to be communicated were different. With the invention of ALOHA and ARPA in early 1970s [1], these fields started to integrate. The same machines were now able to store, process and send data—with the involvement of some Data Communication Equipment (DCE). Today, we see an almost complete integration. A normal computer can store the data of any type—voice, video, images, programs, games, etc.—, process it, and send it to another computer. Storing, processing and transmitting use data in compatible formats and take place over computers [2], [3].

Because of this integration, during the last two decades, the use of networks has grown drastically both at the individual and organizational level. Today networks have become an integrated part of human life. Their use ranges from military applications to space science and business applications to home and leisure activities. The tremendous increase in the use of networks everywhere has transformed them into a basic necessity and a very crucial theme for researchers. While we increasingly need networks, we also have increasing problems associated with them. So, we need more research into different concerns relating to the networks—like stability, scalability, delays, bandwidth and congestion etc. Different type of networks exhibit different problems, thus having different themes of research. Ethernet, a switched *Local Area Network* (LAN) is one of the most used

networks. In the following, we introduce Ethernet and present some problems associated with it in the case of *Åbo Akademi* (ÅA) university Network. We then propose some possible solutions justified via simulations.

II. Switched Ethernet

In the late 1960s, Norman Abramson along with his colleagues at University of Hawaii introduced the concept of *contention based networks* [1], [3]. Their network ran on a shared wireless medium with a medium access scheme named ALOHA. The resulting network was called ALOHAnet—a broadcasting network because of the shared wireless medium. In 1972, shortly after the introduction of ALOHA, Bob Metcalfe and David Boggs at PARC, XEROX Corporation further came up with enhancing the original ALOHA concept to a 2.94 Mbps data rate network [1], [4]. The major enhancements were the use of a cable instead of the wireless medium, the introduction of carrier sensing, and the speed. This newly introduced medium access mechanism was called *Carrier Sense Multiple Access with Collision Detection* (CSMA/CD). The resulting network was named Ethernet [1], [2], [5].

From 1972 to 1990, Ethernet received a number of modifications and enhancement but the core concept of the technology remained the same. With the advent of the Ethernet switch at Kalpana [1], [6] in 1990, the shape of the underlying technology and devices changed. The network started to use Ethernet switches instead of hubs. The basic technique in an Ethernet switch was similar to a telephone switch in which the two ports of the communicating parties were connected for the time of the communication only. This relatively new invention for the Ethernet changed the way the medium was being used. With the use of a switch the network was no longer a broadcasting network. Three years after the introduction of switches, Kalpana delivered another big invention in 1993 namely the *full-duplex transmission* mode. The full-duplex transmission allowed the clients to send/receive simultaneously. Before the invention of Ethernet switches, the full-duplex transmission was not possible because the network ran on a shared medium. Later on, in 1997, IEEE ratified its standard 802.3x for full-duplex/flow control Ethernet [1], [7].

These inventions unleashed the realms of Ethernet. Soon after, in 1995, the 100 Mbps version arrived as IEEE802.3u [7], then in 1999, 802.3ab which provides a speed of 1000 Mbps. Then in 2002, IEEE802.3ae was published which further enhanced the Ethernet to 10 Gbps. And now IEEE 802.3ba group is working for the ratification of 40 Gbps and 100 Gbps Ethernet standards [8]. All this rapid development became possible because of the invention of switching technology for Ethernet.

III. Switched Network Design

Several protocols rely on directed broadcasts for their correct functioning. These protocols include, for example, *Address Resolution Protocol* (ARP) and *Routing Information Protocol* (RIP) [9], [10]. As the networks grew bigger, the problems associated with one large enterprise-wide single broadcast domain became apparent in terms of congestion, latency, waste of bandwidth and the increased load on the processors of connected nodes [10] [11]. A traditional solution to resolve the problem was to divide the whole network into several small broadcast domains with the use of a *router*. Hence, routers are CPU-based devices which calculate forwarding and routing decision on a per-packet base. The involvement of a processor and memory for the forwarding/routing of each and every packet and its decapsulation/encapsulation introduces a considerable amount of delay that becomes a bottleneck in high-speed switching LANs [10]. With the advent of *Virtual LANs*, *higher layer switching* and *multi layer switches*, the routers that were being used to separate the broadcast domain are replaced with multi layer switches [12]. But the huge sizes of networks and the large variety of devices and functionalities available has made the network design much more complex [12]. Building flat networks without a proper design strategy is not any more in practice now. Today, designers use what we recognize as the hierarchal network design [13], [10], [14].

A. Hierarchal Network Design

The hierarchal network design is based on the traffic flows in enterprise's *Campus Area Network* (CAN). A CAN is normally a complex network, hence for the simplicity it is decomposed into three layers. First, we have the traffic which may be destined for services which are campus-wide or external in their scope. Second, there is the traffic which is between similar broadcast domains on the same CAN. The third and the basic one is destined for a local broadcast domain [10].

To clarify this structure we take an example of Åbo Akademi that is composed of several faculties with each faculty having several departments. Figure 3.1 illustrates the concept of every department having its own network segment. The traffic generated from a client on the segment of the Department of Information Technologies and destined for the same segment is local traffic. If the source is at the Department of Information Technologies and the destination is at the Department of Chemical Engineering (same faculty) then the traffic is remote. And finally, if the traffic is generated for a destination which is neither in same segment,

nor in the same faculty—may be outside the campus network, or a resource on the campus network—then this traffic is campus-wide or enterprise-wide in its scope.

Based on these traffic flows, a hierarchal network design consists of three layers. The access layer [10], [12], [15], [16] provides the connectivity to the end devices. The medium used by these devices can be in any form, ranging from wired rings, shared or switched mediums to the wireless medium. The devices which form this layer are normally functioning at the data link layer. Logically, the access layer devices need not have any routing or similar network layer functionality. This is because their role is to provide medium access to the end user. The only other functionality they need to perform is the separation of broadcast domain into several smaller broadcast domains. This job can be accomplished very well with the definition of VLANs.

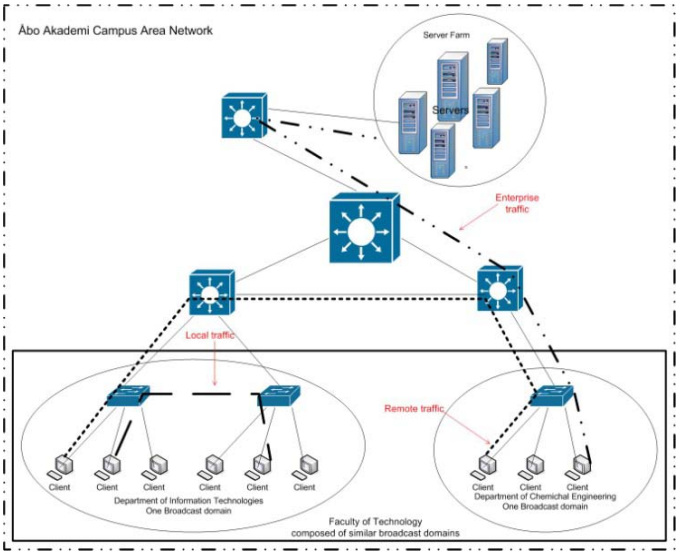

Figure 3.1: An example of different traffic flows

The distribution layer [10], [12], [15], [16] aggregates the traffic coming from the access layer devices, performs inter-VLAN routing and enforces enterprise policies. If the traffic generated by the end systems is destined to the same broadcast domain, then it needs not to be sent to the distribution layer. If the traffic flows need to go out of the generating broadcast domain, the access layer device hands over the traffic to the distribution device. Distribution device checks for all policies, access control mechanisms, and routing information and then acts as defined for that particular type of traffic.

The core layer [10], [12], [15], [16] is the heart of a network. It is the place where the aggregation of the entire traffic over a network takes place. Due to very high volume of traffic present, the devices at the core layer should be extremely efficient. This efficiency requirement is achieved with a combination of two aspects. First, very efficient devices need to be used with the greatest throughput both at network layer and data link layer. Second, this layer is given the responsibility to switch frames on data link layer. Normally, no inter-VLAN routing, packet processing, enforcement of policies or similar functions can be given to this layer—although its devices can perform all these functions.

IV. Åbo Akademi Switched Ethernet

Åbo Akademi University is a traditional seat of learning situated in Turku, Finland. The university owns a reasonably large network which is spanned over the buildings scattered across the city. The network is built upon Ethernet technology using all the switching devices. Only the wireless LAN devices are on a shared broadcasting medium network due to the natural constraints of wireless medium. The network has a hierarchal structure which is apparently built upon three layers. One layer is providing connectivity to end devices. The middle layer is aggregating the traffic from end switches. And the third layer is working as the core of the network. However, these three layers do not fit in the defined three layer architecture that we discussed in the previous section. This is because the logical functions a distribution layer performs—inter-VLAN routing, policy enforcement etc—are not being performed in the middle layer in Åbo Akademi network. This middle layer is merely aggregating the traffic from end devices and passing it to the network backbone. For these aforementioned reasons, we classify Åbo Akademi (ÅA) network as a three layer network having *core*, *aggregation* and *edge* layer—and not the core, distribution and access layer.

A. ÅA Network: The Core Layer

The core layer in ÅA network is the backbone network that provides means for connectivity between distant buildings located throughout the city and to rest of the world. In Figure 4.1, we illustrate the architecture of Åbo Akademi backbone network and its devices.

It is apparent from the figure that Åbo Akademi network comprises a *firewall*, four *routing-switches* and links between them. Three routing-switches are providing connectivity to aggregation layer and firewall, while fourth

routing switch is used to communicate with rest of the world. The description of the objects in Figure 4.1 is as following:

Network Clouds: there are four network clouds. `TuY/TKKK` cloud represents the university of Turku network. The second cloud is `Funet` that provides the gateway to the Internet. The outside world traffic of ÅA network passes through this pathway. The third cloud represents the `Syduast` network. These three clouds are connected over 1000BASE-LX links. Finally, we have the *Demilitarized zone* (DMZ).

Routing-switches: the backbone has four routing switches. The `snoopy-ext-rtr` is the end routing-switch which is providing connectivity to rest of the world. The other three routing switches form a ring in between them. This ring provides connections to all the aggregation layer devices in ÅA network.

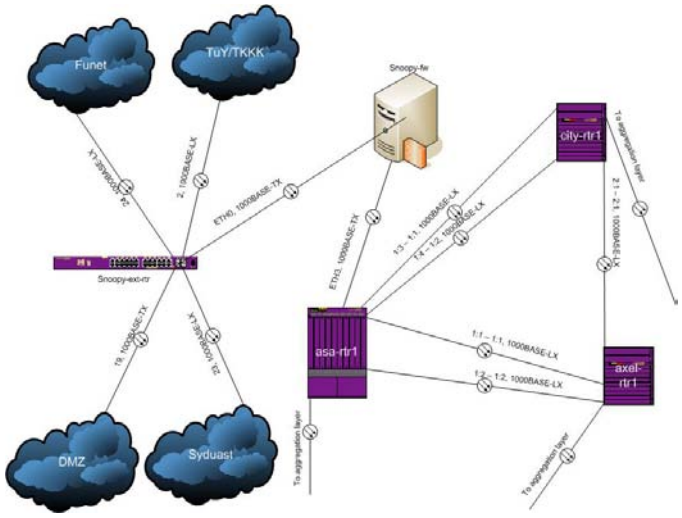

Figure 4.1: ÅA network backbone

Firewall: `snoopy-fw` is a heavy duty server computer that runs checkpoint firewall software to implement organizational policies and to protect Åbo Akademi network from malicious network attacks. All the traffic going out to or coming in from the Internet passes through this firewall

B. ÅA Network: The Aggregation and Edge Layers

Every building of Åbo Akademi has a separate aggregation switch with some exceptions when several buildings located in close proximity use a shared aggregation switch. For example, `DataCity` and `Korposträm` buildings do not have their own aggregation switches. These buildings share `BioCity` aggregation switch. All the aggregation switches finally terminate in one of the three ring forming core layer devices named `city-rtr1`, `asa-rtr1` and `axel-rtr1`. On the other side, these aggregation switches provide connections to the edge layer switches. Several edge layer switches together cover an area/building of Åbo Akademi. For simplicity, we do not present the full details of the network here. We refer to the original work for the complete details.

V. PROBLEMS, SOLUTIONS AND SIMULATIONS

Due to the large number of devices and continuous expanding size of the network, ÅA network has accumulated complexity in the structure. We have sought to find out the structural and design problems in the network especially related to efficiency. Upon analyzing the ÅA network we have uncovered four main areas of potential problems: the naming convention, the extended hierarchy, the uplink bottlenecks and the (lack of) fault tolerance. Here we describe our findings, suggest some solutions, and present the simulated results of current and suggested scenarios via OPNET IT Guru Academic Edition. The simulations are presented for all but the first problematic area, as it was not applicable to this.

A. Naming convention

Network device names can either be user-friendly or not. When we talk about user-friendliness, we refer to the ease of utilization. The major service offered by a name is the identification of an object, hence the user-friendliness in names essentially means how easy the identification of an object is, whose name is being addressed. This identification may be in several aspects, for example, identification of hardware, tasks, geographical location, location in network plane etc.

Mainly, network device names can either be arbitrary or containing precise information in clear form or some kind of encoded form. For example, the aggregation switch at `biocity` building of ÅA network is named `bkf-esw2`. This name does not seem to be conveying much information about the type, job and place in hierarchy of Åbo Akademi network.

If we want to be precise in conveying information through names, we have to assign it a name like *`eswbiofloor1-bkf-esw3-city-rtr1.abo.fi`*. Now this name tells us that this is an edge switch

<esw> at Biocity <bio> first floor <floor1> connected to aggregation layer switch bkf-esw3 which belongs to city-rtr1 hierarchy. The problem with this kind of unstructured and non-planned naming scheme is that the information relays to everyone. Network personnel would not like that everyone just knowing the name of a device gets to know about all of its information. The simple solution is to encode this information in a way that only concerned personnel can decode. For this purpose we have to develop a naming convention. An example naming convention may be like the following:

```
<type><building><wiring closet><device number>-<aggregation tree>-<core
tree>.<domain>
```

The elaboration of variables in this naming convention is given in Table 5.1

The name of above mentioned device now becomes *eswty6204103-bct-ct.abo.fi*. Because we have fixed the length and type of variables, the absence of any delimiter in first portion does not create an ambiguity. We can still get that first three characters tell this is an edge layer (esw) switch at Biocity (ty6). The next four digits tell us the wiring closet along with the device number shown from next two digits. After that we used some delimiters. Using delimiters or not using them is just a matter of choice. One may use them throughout the name, and other may design a convention without them.

TABLE 5.1
DESCRIPTION OF NAMING VARIABLES

Name	Length and Type	Description
type	Three characters	It is the type of device, a device can be for example esw: edge switch asw: aggregation switch cor: core switch rou: router
building	Three alphanumeric	It is the name of the building in which the device is geographically located. A table can be created for building codes using either name initials like bc for Biocity or with some other information, for example, the address of Biocity(Tykistökatu 6): the code can then be ty6.
wiring closet	Four digits	It is the number of the wiring closet in which this device is physically located.
device number	Two digits	A wiring closet may have multiple devices. So these devices can be numbered to keep the names unique.
core tree	Two alphabets	ct: tree rooted at city-rtr1 as: tree rooted at asa-rtr1 ax: tree rooted at axel-rtr1
aggregation tree	Three alphabets	Every tree rooted at one of the core layer devices can be sub-divided into small aggregation level trees. These trees can be coded as bct: tree rooted at aggregation layer switch at Biocity. ict: tree rooted at aggregation layer switch at ICT-huset gad: tree rooted at aggregation layer switch located at Gadolinia building
domain	Variable	This is of course abo.fi in our case

Another advantage is that, by fixing the type and length of variables in the name, automated scripts using wild cards can be run very efficiently throughout the network. For example, a script meant to do a task on all the aggregation layer switches in Åbo Akademi network can use asw* as a parameter. Now it will address all the aggregation layer switches (leaving all other devices). Currently, the aggregation switch at Arken is arken-bdk-1-1-esw1, at Gadolinia it is gado-esw1 and at Biocity it is bkf-esw3. With current names there is no mechanism of using wild cards, but instead one has to write down all the names separately.

Having such long encoded names may seem like a questionable choice. However, this approach encodes the device names and achieves the double benefit of hiding obvious information from hackers/crackers as well as displaying necessary information for network personnel. The level of encryption here is a matter of choice. The more difficult the codes are, the tighter the security level is, hence the difficulty in decoding the device information. Therefore, a moderate choice in difficulty seems suitable. For an overview of naming conventions deployed in different university campuses, see [17], [18], [19].

B. Extended Hierarchy

There are several paths in Åbo Akademi network where the structuring goes beyond the conventional three layer architecture: this really increases the complexity of the network structure. Namely, some edge layer switches extend the hierarchy. This introduces a fourth level in the network, connected to edge layer and residing beneath it. On some paths the depths of hierarchy goes to even fifth level. The network segment under gripen-esw1 switch is one such example. In Figure 5.1, we show the depth of hierarchal structure under gripen-esw1.

Extending the hierarchy in this way is a bad network design practice. A good network design should not go beyond a third level [20]. The more processors, queues, memories and processing one gets involved in the path, the more increasing the delays. If an extension is needed, i.e. a new switch has to be added to the network, it should be given connection from an aggregation layer switch, instead of an edge layer switch.

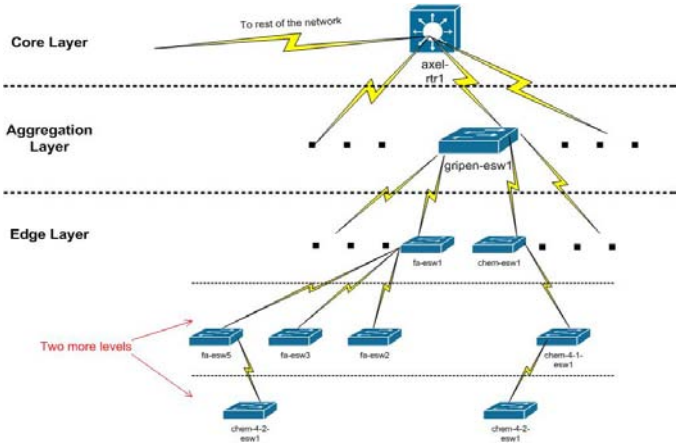

Figure 5.1: Extended hierarchy under `gripen-esw1`

The aggregation layer switches under which this kind of structure has been build in the Åbo Akademi network are `gripen-esw1`, `gado-esw1`, `bib-esw1`, `humanisticum-esw1`, `domus-esw1`, `axel-esw1`, `bkf-esw3` and `ict-bd002-esw1`. This comprises a major part of Åbo Akademi network's aggregation layer.

Suggested Solution, Simulations and Results

In Figure 5.1, we have shown a part of the Åbo Akademi network where the hierarchy goes to a depth of fifth level. We have simulated this model in OPNET IT Guru with two scenarios, one showing the original hierarchy, and the other built on a proposed strict three level hierarchy. The second scenario which is built for a strict three level hierarchy places switches of 4th and 5th level directly under `gripen-esw1`. The purpose of this simulation is to verify that increasing the depths of hierarchy induces unnecessary delays in network traffic. Figure 5.2 shows the models built for this purpose.

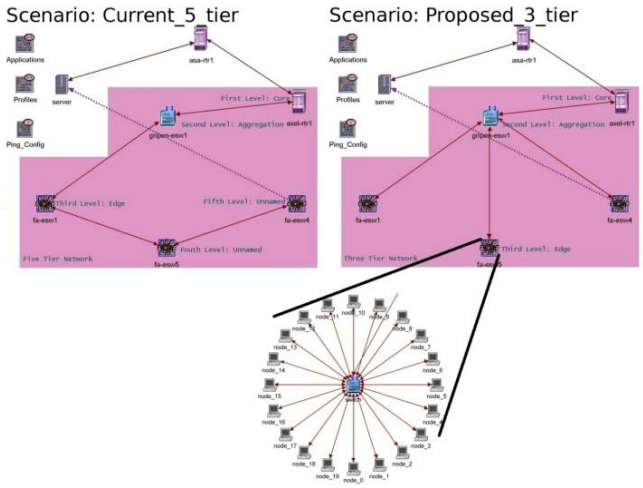

Figure 5.2: Extended hierarchy: simulated scenarios

In the left part of Figure 5.2, the original hierarchy is shown. On the right hand, the proposed hierarchy is modeled, namely a 3tier hierarchy.

On the top, two routing switches named `asa-rtr1` and `axel-rtr1` are installed to build up a core layer. These are BlackDiamond 6808 and Alpine 3808 respectively. The device models chosen are the same as installed in the real Åbo Akademi network. Then we have an aggregation layer switches named `gripen-esw1` which is an HP ProCurve 4000M. Then there are edge layer LAN segments comprising three switches and 20 client computers connected to each switch. There is a server which is connected to `asa-rtr1`. This server computer is receiving requests from the clients on LAN and sending the replies back. All the links in this simulation are 1Gbps, except the links between workstation and the switches. Links are configured without any background link utilization to find out the maximum effect of extending hierarchy. If we find the delays in this scenario where links are totally free—there is no background utilization—then there would be more delays when links would be in heavy usage.

The objects `Applications`, `Profiles` and `Ping_Config` are used to define the type, flow, timing and amount of traffic. The object `Applications` actually defines the applications which are in use on this simulated network. It can be used to define a number of applications. For this, and the others simulation in next sections, we have chosen a few applications randomly. The object `Profiles` defines the way an application is used to generate traffic by this particular scenario. And the object `Ping_Config` is used to define ping packets generation and the behavior of the ping program, i.e., the interval between packets and repetitions etc. There is

also a direct link between `fa-esw4` segment and the `server`. It is not a physical link. It is instead a logical link dedicated to defining the flow of the ping traffic as specified by the object `Ping_Config`.

After running these scenarios for 30 simulation minutes, we gathered the statistics for three different values namely Ethernet delay, ping response time and Database entry response time. These statistics are gathered for an end system `node_1` on `fa-esw4` segment. They show the difference in above mentioned statistics for when `fa-esw4` is directly connected to aggregation layer, and when it is connected to `fa-esw5`, which connects to aggregation layer via `fa-esw1`.

Ethernet Delay

The difference in Ethernet delay in both cases is significant. The delay figures are given in milliseconds that are not insignificant because engineers are researching ways to reduce nanosecond delays [21].

In Figure 5.3, we show the difference between a 3tier and a 5tier topology.

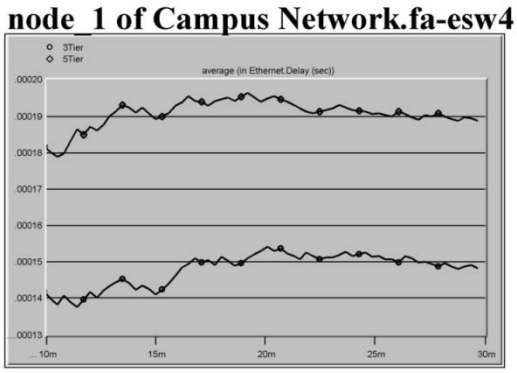

Figure 5.3: Extended hierarchy: Ethernet delay

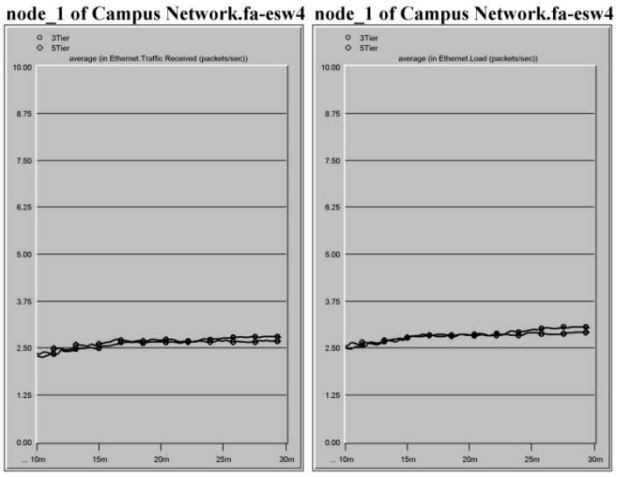

Figure 5.4: Extended hierarchy: Ethernet traffic and load

The delay in Ethernet traffic in the case of a 3tier topology is measured around 0.15 milliseconds on average and it keeps around 0.19 milliseconds in the case of a 5tier topology. One thing to notice here is that the traffic load and the received traffic for both topologies is configured to be the same. We illustrate this similarity in Figure 5.4.

Ping Response

The induced delays due to building an extended topology are also exhibited in the ping response time. The average delay for the 3tier topology is around 0.29 milliseconds, while the average delay in the ping response for 5tier topology is almost 0.39 milliseconds. We show the collected statistics in Figure 5.5.

The ping traffic generated by the two scenarios is same. In Figure 5.6, we show the counts of ping request and ping reply packets exchanged between the server and the client.

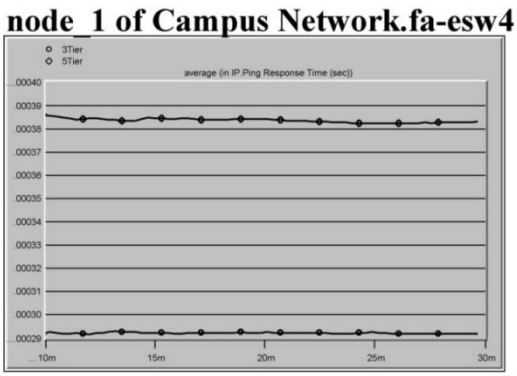

Figure 5.5: Extended hierarchy: ping response time

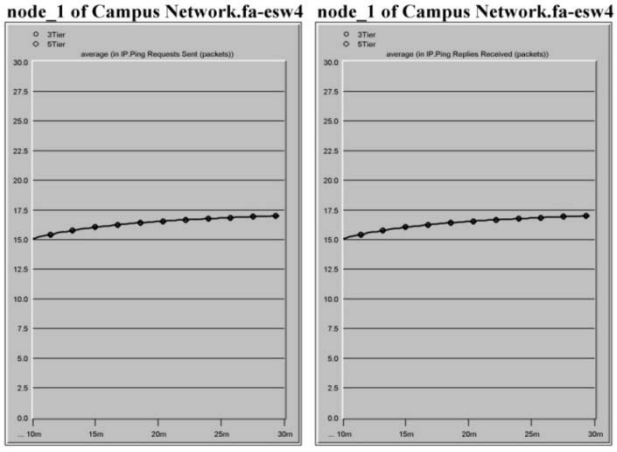

Figure 5.6: Extended hierarchy: ping traffic

Database Entry Response

The extended hierarchal structure also affects the performance of database entry packets. In Figure 5.7, we show the differences in the proposed and the current structure. The response time for a 5tier structure is 160 to 170 milliseconds. On the other hand, if `fa-esw4` is directly connected to aggregation layer, this response time drops to around 140 milliseconds. The difference of 20 milliseconds is a very big figure in terms of network response time.

The traffic sent and received for both scenarios is almost similar which keeps around 0.03 to 0.04 packets per second. We show this traffic patterns in Figure 5.8. The minor difference between the two traffic flows is not significant as it is less than one packet per second. The difference comes due to the generation of traffic based on the pseudo-random generators. Otherwise the configuration of the traffic for both scenarios is the same.

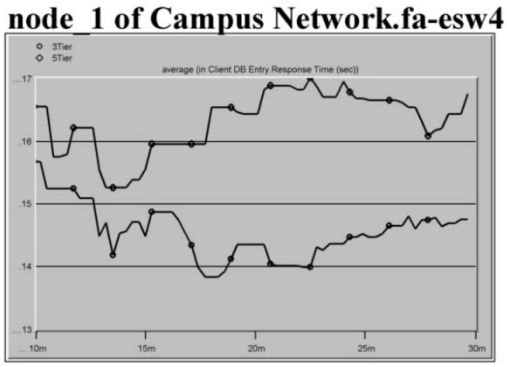

Figure 5.7: Extended hierarchy: DB entry response

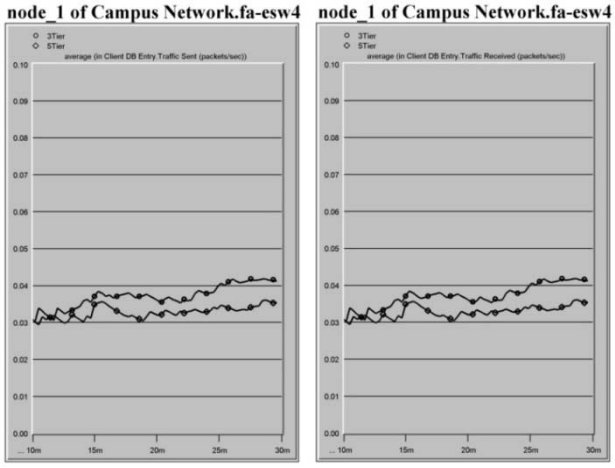

Figure 5.8: Extended hierarchy: DB entry traffic

It is very clear from the figures generated by the simulations that extending the hierarchy to a higher level induces unnecessary delays in network traffic. Therefore, the unnecessary extension of hierarchy should be avoided.

C. Uplink Bottlenecks

The concept of network traffic aggregation refers to the act of collecting several small links to a bigger link. Ideally, if one has to aggregate x network links of y Mbps capacity each, then the aggregating link should be capable of carrying $(x \times y)$Mbps data. However, not all the links are fully busy all the time and consequently, a general rule of aggregating the network traffic is to use the ratio of 3:1 [13] (e.g. 30 links of 100 Mbps each are aggregated into one link of 1000 Mbps, instead of one link of 3000 Mbps). Technological and economical limitations do not allow us to provide uplinks that aggregate the full capacity of a network segment. What the designers do is provide the best solution which falls in between cost effectiveness and quality effectiveness.

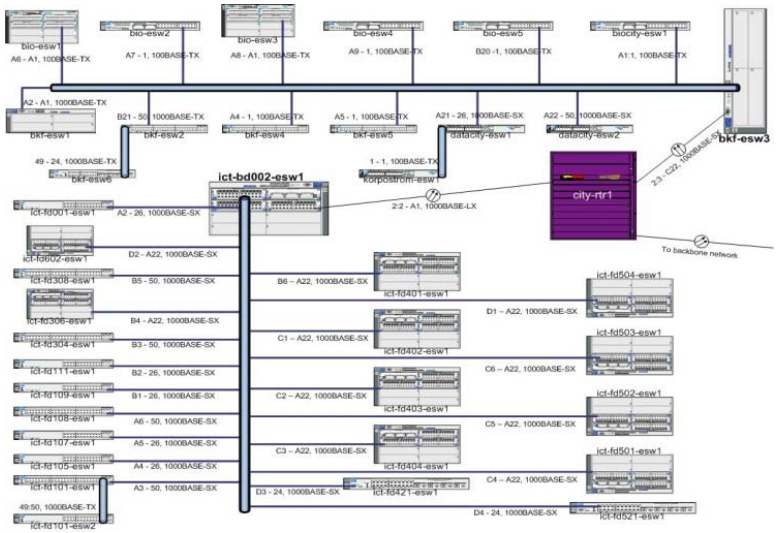

Figure 5.9: ÅA Network: city-rtr1 hierarchy

Åbo Akademi network at some places does not aggregate the links at all. The links being aggregated and the link which is aggregating are almost of same capacity. This situation simply creates a bottleneck. Consider a

network segment of Åbo Akademi that we present in Figure 5.9. All the edge switches in `ICT-huset` are connected with aggregation layer switch `ict-bd002-esw1` over a link of 1Gbps. This aggregation layer switch is further connected to `city-rtr1` over a link of 1Gbps bandwidth. The traffic from 22 switches is being aggregated at `ict-bd002-esw1` over 1Gbps links. Roughly this becomes 22Gbps capacity which is being aggregated and then transferred to `city-rtr1` over a link of 1Gbps. This presents a very high potential for bottlenecks.

The solution to the problem is providing enough capacity in the aggregation links. If the clients are connected to edge switches over 100 Mbps, then the edge switch should be connected to aggregation layer over a link of at least 1000 Mbps, and so on. The backbone should be running at least ten times the speed of end stations [20]. We now present what we have simulated as the suggested and the current scenarios for a segment of ÅA network.

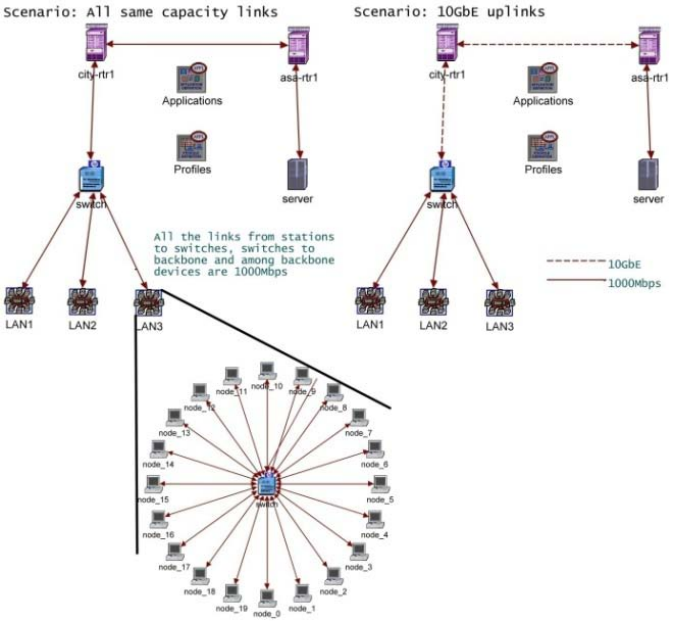

Figure 5.10: Uplink bottlenecks: scenarios

Suggested Solution, Simulations and Results

To show that an uplink of the same capacity as the links to end nodes creates a bottleneck and reduces overall network efficiency, we simulated a network segment with two scenarios. The first scenario simulates all the links with the same capacity. The second scenario simulates uplinks with a bigger capacity as we have shown in Figure 5.10. On the left part, the scenario is similar to the current structure of Åbo Akademi network. All the links either connecting end stations or providing uplinks are of 1Gbps capacity. On the right part in the Figure 5.10, the proposed architecture is simulated. In this proposed architecture, the end stations are connected to switches over a link of 1Gbps capacity and the switches are connected to upstream switches over links of 10Gbps capacity. The objects `Applications` and `Profiles` are used to configure applications for end stations. The end stations generate email and ftp requests destined for the server.

After running this simulation for 30 simulation minutes, we have collected statistics for queuing delay, ftp traffic and the email traffic.

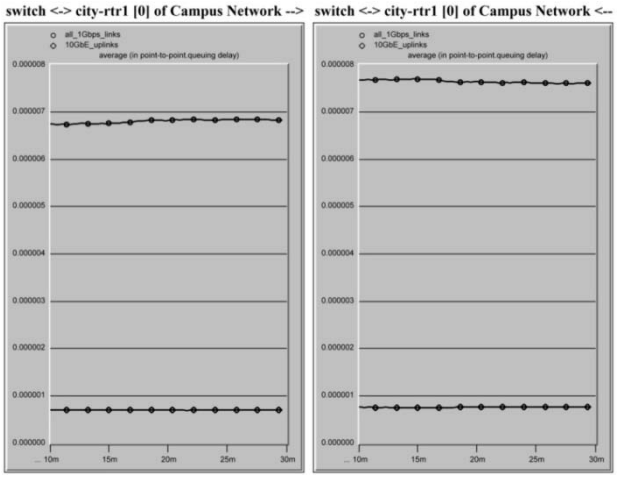

Figure 5.11: Uplink bottlenecks: queuing delay

Queuing Delay on Links

The term Queuing delay refers to the time a chunk of data waits in a queue for its turn for transmission [22]. In Figure 5.11, we show the differences in Queuing Delay of two simulated scenarios. It is measured on the uplink between devices `switch` and `city-rtr1`. The difference in queuing delay of both scenarios is apparent in Figure 5.11. The proposed scenario has a queuing delay which is less than one microsecond, while the current scenario exhibits a queuing delay which keeps around 7 microseconds.

Email Response Time

Because of the same `Profiles` and `Applications` objects used in both scenarios, the email traffic sent and received in both cases is the same. We have illustrated this in Figure 5.12. However, there are still differences in the response times of both scenarios, due to the reason that in first scenario the uplinks create a bottleneck. In Figure 5.13 and 5.14, we show the results collected from the simulation. The email upload response in the scenario with bigger uplinks is around 1.025 milliseconds, while it is around 1.125 milliseconds in the scenario with same uplinks. Similarly, the email download response time is around 1.025 milliseconds for the scenario with bigger uplinks, while it keeps around 1.1 milliseconds for the scenario with same uplinks.

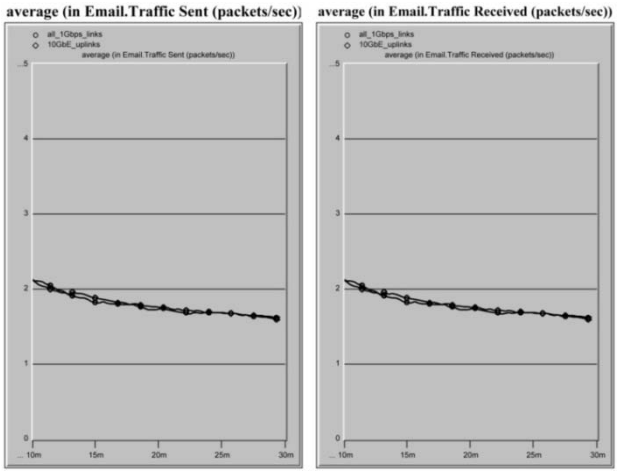

Figure 5.12: Uplink bottlenecks: email traffic

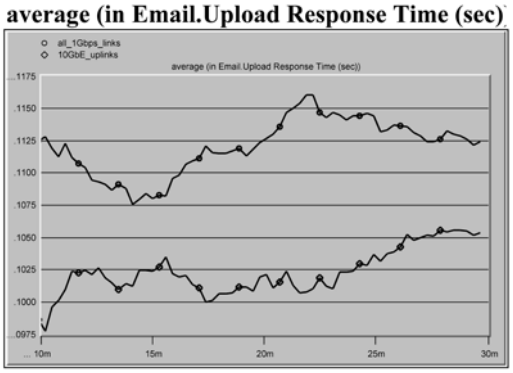

Figure 5.13: Uplink bottlenecks: email upload response time

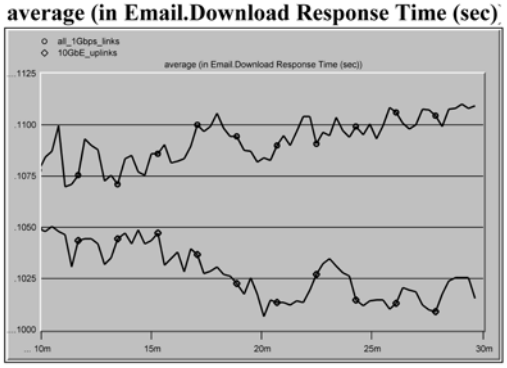

Figure 5.14: Uplink bottlenecks: email download response time

Ftp Response Time

The differences in both the upload and the download response time in ftp traffic are very significant. We show the difference in download response time in Figure 5.15. The response time of the two scenarios almost keeps 5 to 10 milliseconds of difference all the time. This difference is very noticeable and should be eliminated. Similarly, Figure 5.16 shows the difference of response time in ftp upload. The same behavior is exhibited here: the difference keeps between 5 to 10 milliseconds. The traffic for both scenarios is generated by the same `Profiles` and `Applications` objects, hence it is the same. We have illustrated this in Figure 5.17.

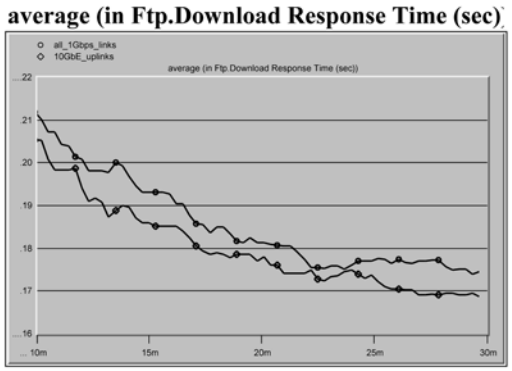

Figure 5.15: Uplink bottlenecks: ftp download response time

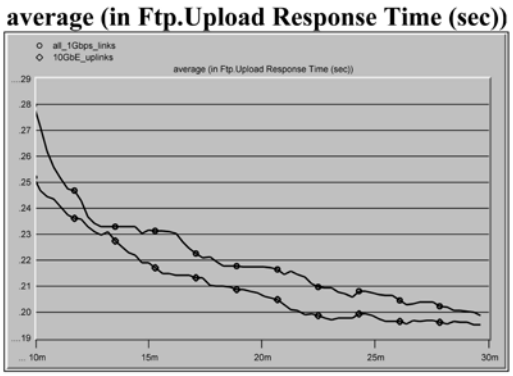

Figure 5.16: Uplink bottlenecks: ftp upload response time

D. Fault tolerance and Redundancy

Essentially, fault tolerance refers to the ability of a network to resist against the various kinds of problems ranging from hardware failures to virus attacks and hacking attempts [23]. It is one of the most crucial considerations of a communication networks [24]. To introduce fault tolerance in a network, the first thing to do is identifying single points of failure in the network. A *single point of failure* is a point in a network whose failure can bring the whole network down. These single points of failure are eliminated mostly through redundancy to keep the network from going down in case of any fault occurrence at this single point of failure [25]. This makes the network very much transparent namely to the users, if a fault occurred at a single point of failure where no redundancy was deployed, it will bring the whole network down, disconnecting all the users from their communications. While, if redundancy was deployed at that point, the communication will automatically shift to the redundant resources, leaving user's communications intact and unaware of any failures.

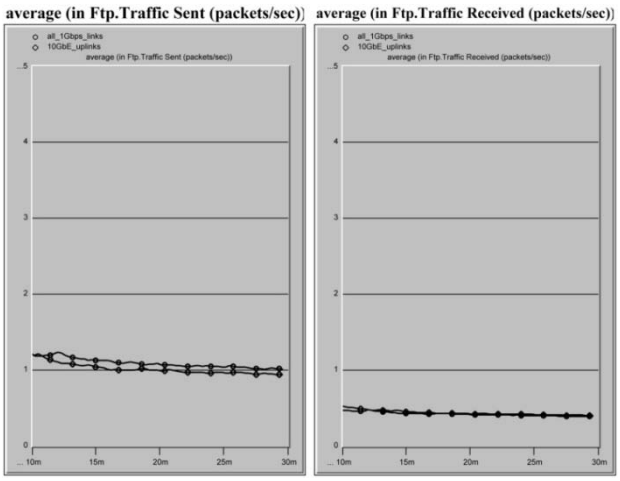

Figure 5.17: Uplink bottlenecks: ftp traffic

In practice, it is not sufficient to eliminate the single points of failure for the entire network. In addition, other points of failure also need to be eliminated for keeping a major part of the network from going down. Consider the failure of `city-rtr1` in Figure 5.9: this would not let the whole Åbo Akademi network down, but still it would disturb a major portion of the network and its users. Hence, it would be useful to eliminate this point of failure as well as other points of failure like this in the network.

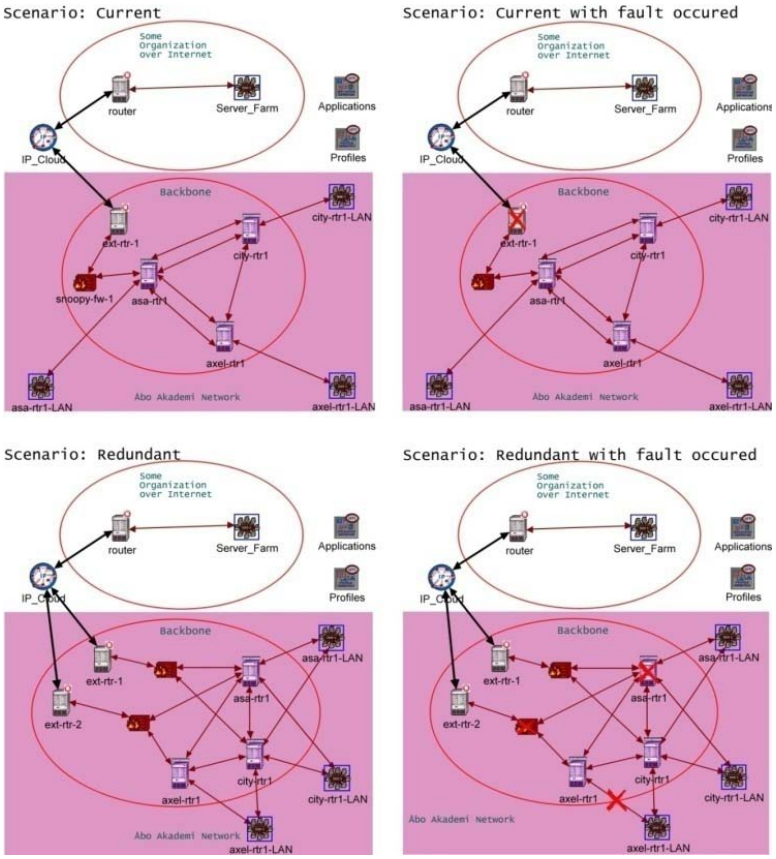

Figure 5.18: Simulation scenarios: fault tolerance

Åbo Akademi network has only one place in the entire network where redundancy is deployed. This redundancy is in links that connect `city-rtr1` and `axel-rtr1` to `asa-rtr1`. There is no redundancy at devices at all; there is no redundancy at links elsewhere at all. Practically, there is no fault tolerance in the network. The de facto solution to this fault tolerance problem is creating a partial mesh [20].

Suggested Solution, Simulations and Results

In Figure 5.18, we illustrate out suggested solution for the problem of fault tolerance with redundant devices and a partial mesh of links between them. The two upper scenarios in Figure 5.18 are based on the current

situation of the network. In the left part, the scenario simulates the situation in which there is no failure in the network. In the right part, the scenario is showing the state when there is a fault occurrence at `ext-rtr-1` device. In the lower two scenarios, the suggested solution is implemented with no fault occurrence at left, and with simultaneous fault occurrence at two devices and one link on the right side.

All of the four scenarios contain similar devices and links between them. The configuration of `Applications`, `Profiles` and thus the traffic is the same. The simulation was run for a total duration of 30 minutes. We now show the result of these simulations.

Effect of Failure on Devices

The first two scenarios that model the current network show the effect of the failure of a device. The last two scenarios show the effect of the failure of two devices and one link. In Figure 5.19, we show the counts of IP traffic sent and received through the edge router in all four scenarios.

In the case of current scenario without a fault the traffic sent and received is around 225 packets per second. But when a fault occurred in the current scenario, the traffic sent and received goes to zero packets per second. It shows that the entire network is no more able to communicate with the Internet. One the other hand, in suggested scenarios, when there is no fault, the traffic sent and received is just the same as in the case of current network without a fault occurrence. But as soon as a fault occurs in the suggested scenario, the traffic drops to around 150 packets per second. This is due to the reason that in the case of a fault occurrence, the devices have to converge once again and calculate new paths. During this time, the communication hangs. But still this process is transparent to the edge devices.

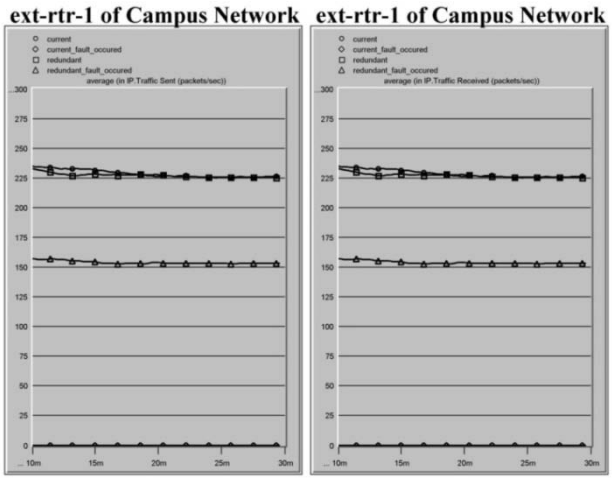

Figure 5.19: Fault tolerance: IP traffic

Effect of Failure on Links

The effect of the failure on links in the current network is fatal. If a device goes down the main links become idle. In Figure 5.20 and 5.21, we show the effects of the failure of `ext-rtr-1` on the link between `ext-rtr-1` and the `ip_cloud` which is representing the Internet. The figures illustrate that in the case of a fault in the current scenario, the utilization and the throughput of the link becomes zero, as there is no path for the traffic. On the other hand, if the fault occurs in the suggested scenario, the utilization and throughput goes slightly down, because of the aforementioned fact that the network needs to converge again. During this time—which is a matter of fraction of seconds—the communication is disrupted. So the overall average utilization and throughput of the links goes down.

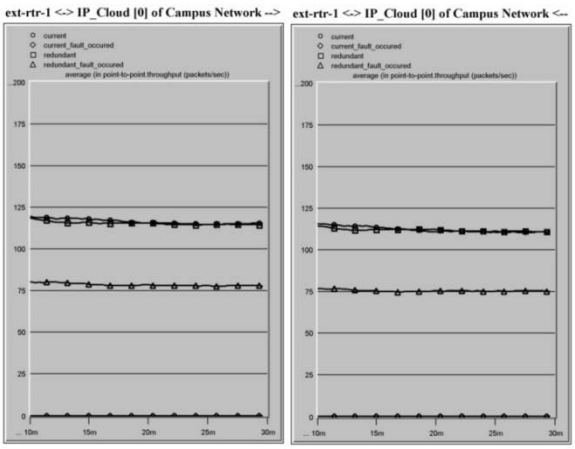

Figure 5.20: Fault tolerance: throughput of link

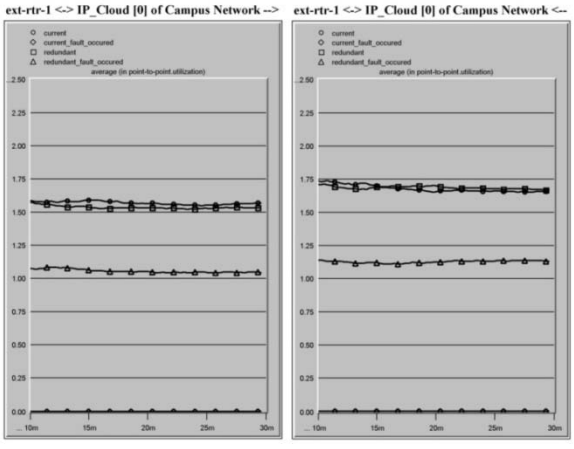

Figure 5.21: Fault tolerance: utilization of link

Effect of Failure on Overall Network Performance

We have recorded the statistics for email applications in these scenarios. In Figure 5.22 and 5.23, we illustrate that there are no statistics for the current scenario with the occurrence of a fault. This is due to the reason that the current scenario has no tolerance against fault.

On the other hand, if a fault occurs in the suggested scenario, the response time becomes almost equal to the current scenario. The response time of current scenario with no fault and the suggested scenario with fault occurred is the same because both scenarios are now running on a single link and a single connecting device to the Internet. While, in the case of the suggested scenario with no fault occurrence, the response time is less because there are two connecting links and devices to the Internet. Due to this fact, the performance of the application is better.

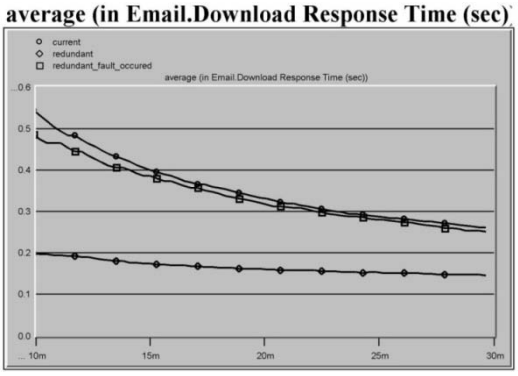

Figure 5.22: Fault tolerance: email download response time

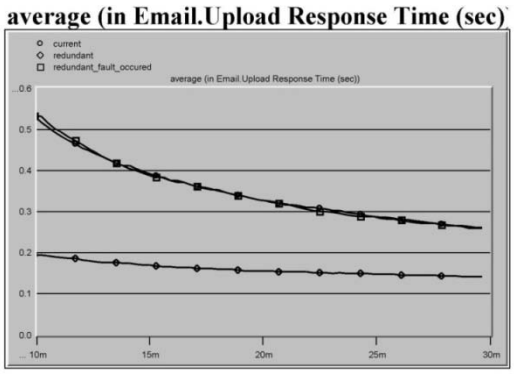

Figure 5.23: Fault tolerance: email upload response time

VI. Proposed Network Design

We have presented in the previous sections that Åbo Akademi network has several problems including the naming convention, the extended hierarchy, the uplink bottlenecks and the (lack of) fault tolerance. Here we propose a backbone architecture that can solve these problems.

The first aspect is the solution of bottlenecks. The proposed solution's entire backbone is built on a 10GbE technology. The second aspect is addition of fault tolerance. This architecture proposes a partial mesh topology to eliminate the points which can bring the whole network down in case of their own failure. A new firewall structure is proposed to eliminate the bottlenecks and single point of failure created by the current firewall setup. A new external router is added to eliminate the chances that the failure of a single external router would cut off the network from the rest of the world. We illustrate this design in Figure 6.1.

Fault tolerance is not the only aspect that is to be acquired through redundancy. Redundancy is also used to enhance the performance. Current network standards and protocols allow us to use redundant device to balance network traffic load. For example, the network load can be equally shared between both of the firewalls, and if one firewall is down, the other can alone handle all the network traffic coming in and going out.

As we have noted before, current Åbo Akademi network at several places violates the conventional layered network architecture by extending the hierarchy to fourth or fifth level. This is totally unacceptable according to a designer perspective. The second major problem at

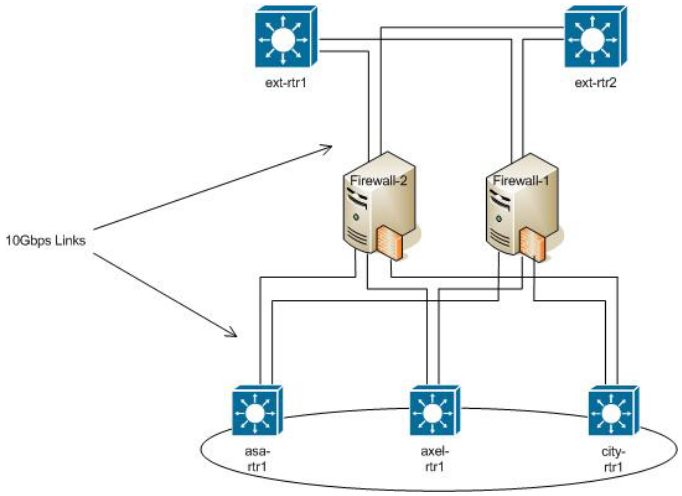

Figure 6.1: proposed backbone architecture

aggregation layer in the current network is the lack of fault tolerance. In Figure 6.2, we present the proposed aggregation and edge layer architecture that can solve these problems.

In the proposed design, the failure of a single link or switch at aggregation layer can be of no harm to the network. It would be even unnoticeable by users. It is because if one link fails, there is always another link to keep communication uninterrupted. And if a switch at aggregation layer fails, there is always another connection to another switch. The only points of failure left in the network are at edge the layer now. These points are not eliminated through redundancy because there scope is very limited. A failure of an edge switch can affect only a number of computers which are connected to it. It can not affect a considerable part of the network, as in the case of a failure at core layer or aggregation layer.

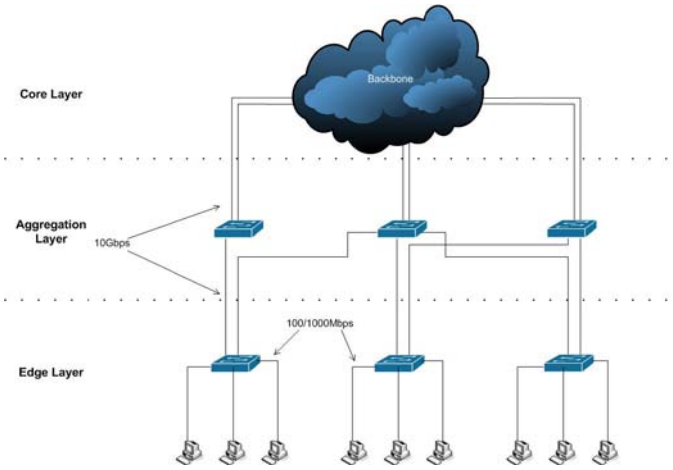

Figure 6.2: proposed aggregation and edge layers

VII. Conclusions and Future Work

The Åbo Akademi network is a switched network built on Ethernet technology. It was established many years ago and had considerably grown during the past years. Our analysis in this work shows that the network has accumulated some complexity. We can speculate about the reasons contributing to this complexity, such as arbitrary growth, limited resources, lack of stringent security needs, and of course, the fact that its main skeleton was designed many years ago. The network's problems revolve around the issues like switched network architecture, aggregation of bandwidth, the extension of hierarchy, bottlenecks, lack of fault tolerance, naming conventions and the old technology backbone. All of these constitute obstacles for having an efficient network.

Our suggested solution and the results of the simulations justify that the backbone should be upgraded to meet the requirements of the new resource hungry applications. Second, further depths in the hierarchical structure currently exist. Simulations and theory show that these further depths in the hierarchal structure exhibit efficiency problems in switching networks. So, the suggested solution is to remove these extensions and connect the hierarchy extending switches directly to the aggregation layer.

One of the biggest problems in the network is the lack of fault tolerance. We have suggested a partial mesh topology with the addition of some devices to introduce fault tolerance. Simulations of the suggested scenario show that our proposed solution does not only introduce fault tolerance but also introduces performance enhancement in the network. Along with this, the presence of a single firewall at the end of the network is another point to reconsider. This device is a dangerous single point of failure in the network. It is also creating a bottleneck because all the traffic from inside the network destined to outside world has to pass through it. We have proposed to install another firewall device. This will facilitate the network with better performance because of the load sharing between two firewalls and with fault tolerance as well, in the case of a device failure.

Another problem is the absence of a structured naming convention. This problem is very crucial on the administrative side of the network. It makes the network more person dependant. We have suggested some naming conventions which can be deployed in the network. Some examples of such naming conventions are quoted from different universities networks.

On the whole, we conclude that there are several problems in the Åbo Akademi network. Some of these problems are very crucial in their nature and demand an immediate solution. We have also proposed solutions to the identified problems and have justified our suggested solutions via simulations.

Future Work Besides the suggested solutions for improving the efficiency of the ÅA network, we see future work on the topic more on the logical level. We plan to investigate the real traffic patterns being originated from the edge of the network as well as the traffic circulation in the network. Analyzing this type of information would allow us to gain a more concrete insight of the bottlenecks and problems in the network.

Along with this, we intend to work on the VLAN architecture deployed in the network. The major issues in this direction would be the definition points of VLAN as well as the inter-VLAN routing. This study will give us important insight into bringing the network to a stricter core-distribution-access type of design.

REFERENCES

[1] R. Breyer and S. Riley, *Switched, Fast, And Gigabit Ethernet* , 3rd ed. , USA : Macmillan Technical Publishing , 1999.
[2] W. Stallings, *Data and Computer Communications*, International ed. New Jersey, USA: Prentice-Hall, Inc., 1997.
[3] A. S. Tanenbaum, *Computer Network*, 3rd ed. New Jersey, USA: Prentice-Hall, Inc., 1996.
[4] CISCO Systems, Inc., Internetworking Technology Handbook - Ethernet. [Online]. http://www.cisco.com/en/US/docs/internetworking/technology/handbook/Ethernet.html
[5] F. Halsall, *Data Communications, Computer Networks and Open systems* , 4th ed. USA: Addison-Wesley Publishing Company , 1996.
[6] CISCO Systems, Inc., Internetworking Technology Handbook - LAN Switching. [Online]. http://www.cisco.com/en/US/docs/internetworking/technology/handbook/LAN-Switching.html
[7] IEEE, IEEE Standards for Local Area Network. [Online]. http://ieeexplore.ieee.org/xpls/abs_all.jsp?tp=&isnumber=1002&arnumber=26520&punumber=2565
[8] (2008, Mar.) IEEE, IEEE P802.3ba Objectives. [Online]. http://grouper.ieee.org/groups/802/3/ba/PAR/P802.3ba_Objectives_0308.pdf
[9] W. Stallings, *High Speed Networks: TCP/IP and ATM Design Principles* . New Jersey , USA: Prentice-Hall, Inc., 1998.
[10] T. Boyles and D. Hucaby, *CCNP Switching Exam Certification Guide*. Indianapolis, USA: Cisco Press, 2001.
[11] CISCO Systems, Inc., Internetwork Design Guide - Appendix E: Broadcasts in Switched LAN Internetworks. [Online]. http://www.cisco.com/en/US/docs/internetworking/design/guide/nd20e.html
[12] CISCO Systems, Inc., Internetwork Design Guide - Internetworking Design Basics. [Online]. http://www.cisco.com/en/US/docs/internetworking/design/guide/nd2002.html
[13] J. J. Roese, *Switched LANs: Implementation, Operation, Maintenance* . USA: McGraw-Hill, 1998.
[14] CISCO Systems, Inc., Internetwork Design Guide - Introduction [Internetworking]. [Online]. http://www.cisco.com/en/US/docs/internetworking/design/guide/nd2001.html
[15] K. D. Stewart III and A. Adams, *Designing and Supporting Computer Networks* . Indianapolis , USA: Cisco Press , 2008.
[16] W. Lewis, *Multilayer Switching Comapnion Guide* . Indianapolis, USA: Cisco Press , 2003.
[17] University of Waterloo, Network device naming standard. [Online]. https://strobe.uwaterloo.ca/~twiki/bin/view/ISTNS/NetworkDeviceNamingStandard
[18] Rutgers: The State University of New Jersey, Rutgers: Telecommunications Division. [Online]. http://www.td.rutgers.edu/documentation/Reference/RUNet_Network_Device_Naming_Convention/index.html
[19] University of Maryland, Baltimore. [Online]. www.umaryland.edu/cits/docs/Campus%20Device%20Naming%20Conventions.doc
[20] J. Swartz and T. Lammle, *CCIE : Cisco certified internetwork expert : study guide*. San Francisco , USA: Sybex, 2001.
[21] D. Sadot and I. Elhanany, "Optical switching speed requirements for terabit/second packet overWDM networks," *Photonics Technology Letters, IEEE*, vol. 12, no. 4, pp. 440-442, Apr. 2000.
[22] J. Davidson, J. Peters, and B. Gracely, *Voice over IP fundamentals*. Indianapolis, IN, USA: Cisco Press , 2000.
[23] S. Mueller and T. W. Ogletree, *Upgrading and Repairing Networks*, 4th ed. Indianapolis, IN, USA: Pearson Education , 2003.
[24] S. Odom and H. Nottingham, *Cisco switching black book*. Scottsdale, AZ, USA: Coriolis Group Books, 2001.
[25] S. M. Ballew, *Managing IP networks with Cisco routers*. Sebastopol, CA , USA: O'Reilly & Associates, 1997.