Automatic Cut-Through Paths

Research Project 2
System and Network Engineering
University of Amsterdam
2005 - 2006

July 4, 2006

René Jorissen
rjorissen@os3.nl

Lourens Bordewijk
lbordewijk@os3.nl

Preamble

This report is written for Research Project 2 for the Master's education in Systems and Network Engineering at the University of Amsterdam. This paper has been written iteratively by the authors. We would like to thank Radia Perlman and Joe Touch for their help with TRILL and RBridges, Marco Ruffini and Donald O'Mahony for their information on Optical IP Switching, Leon Gommans for his expertise in layer two networks. We would like to thank Cees de Laat as research coordinator and Eelco Schatborn for reading the paper. We would like to thank the Amsterdam Internet Exchange for giving us the opportunity to finish our research project. Especially the AMS-IX engineers for their expertise, Elisa Jasinska for providing information about sFlow. A special thanks to Henk Steenman for his accompaniment and devotion during the research project.

Abstract

This paper provides insight to the topic of cut-through paths in a layer two network. Cut-through paths make it possible to switch traffic more efficiently through the network and lessen the load on certain network components. The paper describes the encountered problems, protocols and technics and solutions with regard to the automatic creation of cut-through paths within the AMS-IX network. A interim cut-through solution could be implemented after further research, RBridges are another solution to switch layer two traffic more efficiently in the near future. This latter group depends on the development of key hardware and software components. Cut-through paths will probably be obsoleted through more efficient technology.

Contents

List of Figures

List of Tables

1 Introduction

At the moment more and more organizations and applications require lots of bandwidth, low latencies and low delays. The explosion of Internet traffic has strengthened the need for high-speed backbone networks. The use of shortest and most ideal paths is required to provide high quality and cost-effective services.

The Amsterdam Internet Exchange is one of the biggest Internet Exchanges in the world. Currently they are using a hub & spoke topology to connect their customers with each other. In fact the hub & spoke topology is build redundantly for failover purposes. With the hub & spoke topology, all network traffic has to travel through the core, which results in an enormous impact on the core switch. This is not the most ideal situation in heavily loaded networks like AMS-IX.

This report tries to describe how cut-through paths can be created automatically within the Amsterdam Internet Exchange hub & spoke topology. Cut-through paths are used to forward traffic flows more efficiently through the network. We cannot cover all possible solutions within the four weeks of the project, but we try to give a good overview of possible solutions and points of interest for the current situation.

Chapter 2 describes the problem definitions, which arise when implementing cut-through paths in a layer two network. An overview of the current AMS-IX network can be found in chapter 3. Chapter 4 describes cut-through paths and triggers used to establish a cut-through path. RBridges are discussed in chapter 5. Possible solutions with regards to photonic switches can be found in chapter 6. Chapters 4, 5 and 6 also describe possible drafts for implementation within the AMS-IX network. Security issues can be found in chapter 7. We conclude in chapter 8. Chapter 9 tells more about future research and points of interest on the topic. References and Appendixes can be found at the end of the paper.

2 Problem definition

There can be some confusion about the term *cut-through switching*. This paper is about creating automatic cut-through paths. A traffic flow, which reaches a certain threshold, triggers the creation of a cut-through path. Specific traffic flows are redirected over cut-through paths and the frames are switched over the newly created path. Switching frames over a cut-through path, is called *cut-through switching*.

The term cut-through switching also has another meaning. It is important to determine in which context the term *cut-through switching* is used within this paper. The second, and original meaning from cut-through switching is described by several organizations and people. The following switching method definitions come from Cisco Systems Inc. [27]:

> "Two switching methods can be used to forward a frame through a switch:
>
> - *Store-and-forward switching*
> The entire frame is received before any forwarding takes place. The destination and/or the source address are read and filters are applied before the frame is forwarded. Latency occurs while the frame is being received; the latency is greater with larger frames because the entire frame takes longer to read. Error detection is high because of the time available to the switch to check for errors while waiting for the entire frame to be received;
> - *Cut-through switching*
> The switch reads the destination address before receiving the entire frame. The frame is then forwarded before the entire frame arrives. This mode decreases the latency of the transmission and has poor error detection. Fast-forward and fragment-free are two forms of cut-through switching:
> - *Fast-forward switching*
> Fast-forward switching offers the lowest level of latency by immediately forwarding a packet after receiving the destination address. Because fast-forward switching does not check for errors, there may be times when frames are relayed with errors. Although this occurs infrequently and the destination network adapter discards the faulty frame upon receipt. In networks with high collision rates, this can negatively affect available bandwidth. Use the fragment-free option to reduce the number of collision frames forwarded with errors. In fast-forward mode, latency is measured from the first bit received to the first bit transmitted;
> - *Fragment-free switching*
> Fragment-free switching filters out collision fragments, which are the majority of packet errors, before forwarding begins. In a properly functioning network, collision fragments must be smaller than 64 bytes. Anything greater than 64 bytes is a valid packet and is usually received without error. Fragment-free switching waits until the received packet has been determined not to be a collision fragment before forwarding the packet. In fragment-free mode, latency is measured from the first bit received to the first bit transmitted;
>
> The latency of each switching mode depends on how the switch forwards the frames. The faster the switching mode, the smaller the latency in the switch. To accomplish

> faster frame forwarding, the switch takes less time to check for errors. The tradeoff is less error checking, which can lead to a higher number of retransmissions."

The term *cut-through switching* in this paper is used with regard to switching frames over a cut-through path. A cut-through path is created to lessen the load on the core switch or to optimize the path for a stream between two edge switches. Implementing cut-through paths in a network seems easy, but the contrary is true. Especially when talking about a pure layer two network. In a layer three network cut-through paths can be created more easily, because it is possible to configure a physical loop in the network with the help of routing protocols.

In a layer three network, all routing decisions are made with the help of IP addresses. Routing protocols take care of the routing in the network. A least-cost path is created to a specific destination. Redundant links or loops between nodes are used for failover and load balancing purposes. The routing protocol dynamically converges the network. The time it takes to converge the network depends on the routing protocol used. Another property of a layer three network, and especially of a layer three network component (router), is the ability to divide broadcast domains. A router does not forward broadcast packets received on an ingress interface to a specific egress interface. If a router does not have a path to the destination in its routing table, the router will discard the package.

In a layer two network, this process works differently. A layer two device makes switching decisions based on Media Access Control (MAC) addresses. A switch has a content-addressable memory (CAM) table with MAC addresses and the associated port numbers. If a switch does not have an entry in the CAM table the switch will broadcast the traffic through all ports, except the port where the packet was received. If the switch receives a packet with a source MAC address that the switch does not have in its CAM table, it will put an entry with MAC address and received port number in the CAM table. A switch learns MAC addresses by looking at the source MAC address and the port number where the packet was received.

This learning of MAC addresses is a big problem when implementing automatic cut-through paths in a network. At the moment that a cut-through path is created, a loop will exist in the network. This means that a broadcast storm can be created, because traffic load will increase exponentially in the event of frames with an unknown destination (broadcast traffic). An approach to the problem loops present in a layer two network is the Spanning Tree Protocol (STP). STP has the following definition [1]:

> "The spanning tree network protocol provides a loop free topology for any bridged LAN. The Spanning Tree Protocol, which is also referred to as STP, is defined in the IEEE Standard 802.1D."

STP will provide a loop free tree by blocking one or more paths in the network. This means that with regards to the hub & spoke topology in the AMS-IX network, the newly created cut-through path is blocked, which means that still all the traffic has to travel through the core, or the path to the core is blocked, which means that all traffic has to travel through the cut-through path and from there to the core. Either way, all traffic is concentrated on one or a subset of links. Further is the convergence of STP slow in larger networks, because of the distribution of BPDU's and STP computation.

Given this information, STP is not the ideal solution to handle a loop in the network. To prevent a loop, no MAC addresses should be learned on the cut-through switch port and only traffic intended for the cut-through path should be forwarded over the cut-through path. Also broadcast traffic should not be forwarded over the cut-through path.

A second problem will be the automatic creation and tear down of the cut-through paths. Some type of device, like a server, has to automatically create and tear down the cut-through paths. Cut-through paths should only be created when a certain threshold is reached. This threshold has to be defined based on flow count, packet count and packet size, so an analysis of the network traffic is needed. In the most ideal situation this would be a real-time analysis. The analysis helps by determining when the defined threshold is reached. At the moment that the threshold is reached a automatic cut-through path has to be created. When the cut-through path is created and the threshold (on the cut-through paths) is not reached any more, the cut-through path should be torn down. The traffic analysis, the sampling process and the filtering process are not easy tasks within the AMS-IX network with a average load of 90 Gb/s. Constant analysis of all the network traffic requires a huge amount of storage and CPU performance.

If solutions can be found to this problems, the automatic creation and management of cut-through paths could be a large step forward in providing high speed and cost-effective layer two networks.

3 AMS-IX

The Amsterdam Internet Peering Exchange (AMS-IX) [2] is one of the largest Internet Exchanges in the world. AMS-IX is a neutral and independent not-for-profit Internet Exchange providing services since the early 1990's. With approximately 240 members, including the largest ISP's in Europe and North America. AMS-IX's current average (e.g. June 2006) load [3] is approximately 90 Gb/s with peaks of 150 Gb/s. Even at this volume AMS-IX is continuing to experience exponential growth in data volume, driven by increasingly bandwidth-heavy applications like video streaming.

3.1 Infrastructure

Glimmerglass [6] provided a layer one optical fiber switching solution for AMS-IX that enabled the Internet Exchange to smoothly transition from its four-location, metro ring network topology to a redundant "hub & spoke" topology (see figure 1).

The redundant hub & spoke topology is implemented for failover purposes. A Glimmerglass switch allows AMS-IX to switch between the two redundant networks. At the time of a link failure or another type of problem, this redundant network allows a smooth and easy failover to the second hub & spoke topology.

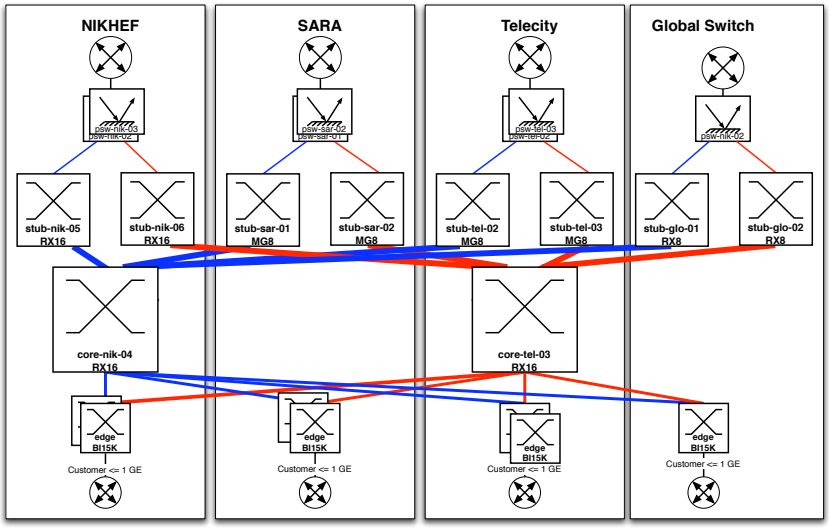


Figure 1: Current topology AMS-IX

The core switches run VSRP (Virtual Switch Redundancy Protocol, Foundry Networks proprietary [4]) to define the active hub & spoke topology and to automatically failover to the redundant topology, based on pre-defined triggers (e.g. link failure). All the edge switches follow VSRP automatically, the Glimmerglass switches follow the VSRP failover based on software developed at AMS-IX.

As core switches (hub) AMS-IX uses the Foundry Networks [5] RX16 systems. There are two types of edge switches. One for customers connecting with 1GE, 100BaseTX or 10BaseT ports and one for customers connecting with 10GE connections. The former group is connected to Foundry Networks BigIron 15000 switches. Each of these is connected to both the core switches. The latter group is connected, through a Glimmerglass System 300 photonic switch, to the

Foundry Networks BigIron MG8 switch. This topology makes it possible for customers to connect to the AMS-IX infrastructure at the following speeds: 10 Mb/s, 100 Mb/s, 1 Gb/s or 10 Gb/s.

The network drawing is not entirely complete. A photonic switch is situated between the core switch and the edge switches. This means that two additional photonic switches should be placed in the figure. The photonic switches are not placed in the figure, because they do not have any special purposes. They are mainly used for management purposes and act like layer one repeaters with regards to customer traffic. The photonic switches between the core and the edges are called *infrastructure* photonic switches and the photonic switches between the 10GE customers and the edge switches are called *edge* photonic switches.

3.2 VLANs

AMS-IX uses different VLANs in their network. The main VLAN contains all the traffic which is related to the regular Internet. Their are separate VLANs for multicast traffic and GRX traffic. About 99% of the traffic is related to the Internet VLAN. VLANs are used the segment a layer two network. So customers in the Internet VLAN can not communicate with people in the multicast VLAN without the interference of a layer three network component (router).

Besides the previously mentioned VLANs, there are also new customer VLANs called "Quarantine VLANs". All new customers are placed in seperate quarantine VLANs for the following reason [7]:

> "AMS-IX defines a fairly strict set of allowed traffic types [8] on the peering LANs. Not all routers (and intermediate L2 devices) adhere to these guidelines; they typically have various protocols turned on by default such as CDP, EDP, STP, DEC MOP, etc., or they present more than one MAC address to the platform. These misbehaving/misconfigured devices potentially endanger the stability of the peers and/or switching platform. Hence, we cannot allow them on the peering LANs.
>
> Rather than act reactively once a member port is in production, we prefer to detect and fix these issues beforehand. Therefore, we introduced the concept of a quarantine VLAN. Once a member's router is connected and the port is up, we can quickly see if it is 'clean' (i.e. adheres to the rules). If it is not, the violating traffic does not harm the rest of the platform."

The different quarantine VLANs function as a sort of security measurement to protect the network and existing customers from new customers with badly configured routers.

3.3 STP

Looking at figure 1 several loops can be distinguished in the network topology. The loops exist because of the redundant hub & spoke topology. Loops in a layer two network should be prevented, because broadcast traffic in a looped network can create broadcast storms. A solution for preventing loops is the use of the Spanning Tree Protocol, as stated in chapter 2: Problem definition. STP would be a solution for the AMS-IX network to prevent loops. Would be, because STP is not used by AMS-IX.

AMS-IX uses the Foundry Networks proprietary protocol VSRP to handle the automatic failover in the event of a link failure. VSRP also helps preventing loops in the AMS-IX network. The

two core switches exchange VSRP Hello messages. These messages contain a VSRP priority per switch. The priority is based on a predefined priority, which is multiplied by the ratio between active backbone links and total number of backbone links. The master and the backup switch are elected with the help of these priorities. After electing master and backup switch, the master switch sends hello messages at a regular interval to tell the backup switch at the master is 'healthy'. When the backup switch does not receive hello messages for a certain time period, the backup switch assumes the role of master switch. This is not the only possibility for a backup switch to take over the role of master switch.

Assume the core switches have a total number of 20 backbone links, which are all active. The predefined priorities are 100 and 101 per switch. The formula for the priority computation is, where Prio = actual priority, abl = number of active backbone links, tbl = total number of backbone links and PPrio = predefined priority:

$$\left[\quad Prio = \frac{abl}{tbl} \times PPrio \right]$$

One core switch will have the priority 100 and the second switch will have a priority of 101. The switch with priority 101 will become the master switch. When one backbone link from the master switch fails, the new priority will be 94. The backup switch now has a higher priority and will become the new master switch.

The loops in the network are prevented, because the master switch will forward all traffic and the backup switch will not forward any traffic and will block all its ports. This means that no traffic can be switched and no MAC addresses can be learned by the backup switch, so no loops exist in the network.

4 Cut-through path

With a cut-through path solution the network capacity will be spread and the performance between the edge switches will be optimized. The cut-through path increases the bandwidth capacity between the edge switches, because the capacity of a cut-through path is larger than the paths going through the core. Several traffic flows are concentrated on the path through the core and the cut-through path is a dedicated connection for one traffic flow. The cut-through path should lessen the load and traffic congestion at the core switches.

4.1 Control Server architecture

A control technology is needed, which automatically and dynamically provisions the cut-through paths. To manage the resources, to collect flow data and to control the dynamic paths, we propose to use the control server architecture in figure 2. The control server makes calculations and automatically configures a dynamic cut-through path. This architecture can manage the resource and traffic information, it considers the priority of all paths and acquires and distributes the network resources effectively.

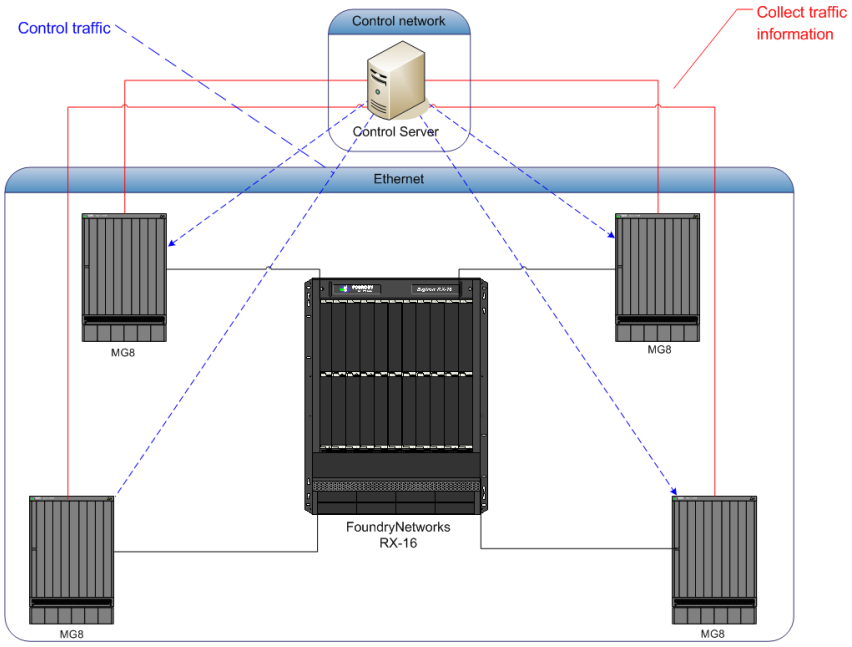


Figure 2: Control Server architecture

4.2 sFlow/NetFlow and resource information

Each time when, at an edge switch, a high traffic volume reaches a certain threshold, the control server must automatically trigger a cut-through path through the network. So the edges should be monitored, to detect high volume traffic flows. The first detection method is provided by the Foundry switches, they have the feature to collect NetFlow and sFlow data without sacrificing network performance. The AMS-IX has only switches that support sFlow, but no NetFlow.

4.2.1 NetFlow

NetFlow is Cisco's metering technology for gathering IP traffic information. NetFlow counts every incoming packet from a port. The information is extracted from the IP header and some pieces of the IP data (TCP and UDP). So NetFlow requires every IP packet to be captured and analyzed. According to Foundry, all network components with the VM1 module should be able to support NetFlow versions 1, 5, and 8.

A trigger option might be collecting a sufficient amount of packets and send the packets in one burst over the newly created path, just like Optical Burst Switching [34]. The best method for collecting traffic information within the AMS-IX network, without affecting the performance and reliability, would probably be through sFlow data.

4.2.2 sFlow

SFlow [9] is a packet-based sampling technology. SFlow gathers information from the layers two to seven. The support for sFlow is built into the JetCore ASIC and the BigIron VM1 Management Module [10]. SFlow can provide information about switch ports, MAC addresses, VLANs, IP addresses and ICMP/TCP/UDP-based and AS-based information.

> SFlow supports high-speed interfaces more easy in comparison to NetFlow, because it does not "touch" every packet for network traffic analysis. "Touchin" every packet at 10 Gb/s, which can deliver a packet throughput of 14.88 MPPS, requires extensive computation.
> Source: Foundry [10]

The produced data volume could be a problem on itself [11]. Packet sampling can handle the volume of high speed backbone links. If no sampling is used NetFlow's result is more complete, but it requires extensive computation. It is faster to take samples, what makes the analysis less reliable. SFlow uses randomness in the process to prevent synchronization with periodic patterns in the traffic. SFlow is not 100% accurate, but sFlow provides a result with quantifiable accuracy. The sFlow accuracy can be estimated with the following formula, where c = number of samples in class:

$$\left[\%error \leq 196 \times \sqrt{\frac{1}{c}}\right]$$

The error percentage decreases by increasing the number of samples per class of traffic, a class of traffic could be voice. Figure 3 shows the error percentage graph for sFlow sampling [12].

4.2.3 Resource information

Foundry's products support SNMP (Simple Network Management Protocol [13]). SNMP makes it possible to query information, like the data/packet transfer per port. Information like CPU utilization, Dynamic Memory utilization, System DRAM Information Group, ARP Tables, System Logging, CAM Statistics and System Process Utilization Table information can also be obtained with SNMP.

AMS-IX's Foundry products allow two features (sFlow and SNMP) to be enabled at once. When correlation of this information is possible, it can be useful in the process by determining the right trigger.

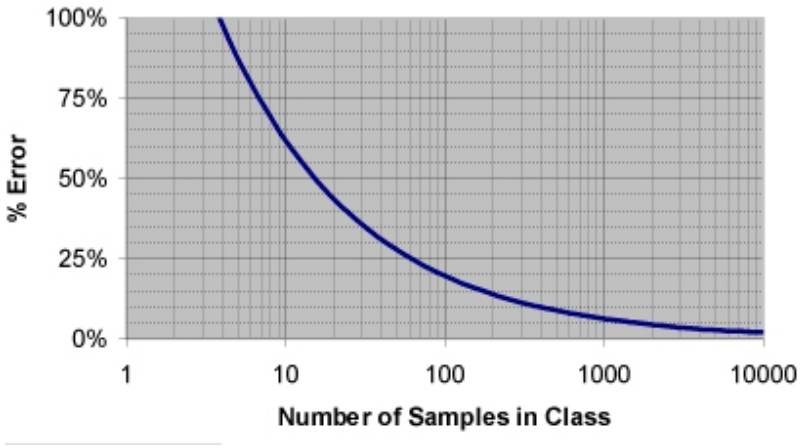


Figure 3: Number of samples in class

4.3 Collection

Working with a central control server, which manages all resources centrally, is preferred when collecting sFlow data. The control server can be used to manage all the network resource information and traffic information. At the time of writing, Elisa Jasinska [14] is doing research at the AMS-IX on how to collect sFlow data from the switches. Her goal is to match the traffic between the edges (customer ports).

SFlow samples packets in a defined rate on the switches, like 1 out of 2048. The sampling rate can be set manually. When the rate is set too low, it could cause inaccurate results. When the rate is set too high, extensive use of memory and processing power could be possible consequences. Elisa Jasinskas research covers the definition of the correct sampling rate.

The samples are send in UDP packets to a "collector" (paragraph 4.1). The collector is a server which decodes the sFlow datagrams. The datagrams can be analyzed with *sFlow Toolkit*, an example of this format can be found in Appendix 9.

The sFlow data will be updated real-time in a database. This data can be used to calculate if it is worth to create a cut-through path. The MAC addresses from the flow data can be used to see from which customer router to which customer router the flow travels. The AMS-IX keeps a XML file with all the MAC addresses from the customers located at each switch port. With this information we know from and to which switch the cut-through paths must be created.

4.4 Trigger

When a certain traffic flow or a concentration of traffic flows reaches a certain threshold, a cut-through path should be triggered. This threshold has to be defined and traffic flows should be measured to compare the amount of traffic with the predefined threshold.

NetFlow could be used to measure the amount of packets on the network. The most entities from a traffic flow could be extracted with NetFlow version 9, which is currently not supported by Foundry Networks. NetFlow version 9 adds a version number, sequence number, time-stamps (indicating the flows begin and end time), number of packets, number of bytes, source and destination IP address, source and destination port numbers, IP protocol, ToS (Type of Service) value and TCP flags to the exported flow. NetFlow does not have the opportunity to

extract MAC addresses from traffic flows. MAC addresses are needed for making forwarding decisions on layer two. The currently used switches do not support NetFlow. This leaves sFlow as the best solution for sampling packets.

Takeshi Yagi et al. [15] propose a method which reduces the number of flows to be calculated. Flows with a high traffic volume get a high priority, because they improve the efficiency of cut-through paths more in comparison to flows with a low traffic volume. The proposed method divides the flows into small groups, sorts them according priority and the number of packets. When dividing the flows in groups, the calculation should be shorter. When the highest priority flows are known, the flows with the highest amount of traffic are also known. Different parameters need to be extracted from the flows to automatically configure a cut-through path.

The AMS-IX network switches a lot of traffic per second, the average traffic is 90 Gb/s. This makes it difficult to start analyzing all traffic, because this would require a huge amount of disk storage. That's why we use sFlow, sFlow has a quantifiable accuracy.

We propose our own sampling process for measuring the amount of flows. The sampling process of these flows is handled per switch. It is not necessary to start the sampling process for a switch, if there's is no need for a cut-through path. This only spoils disk space and CPU power on the control server. So the first step is defining a threshold for starting the sampling process. This threshold could be determined on the switch ports average CPU and bandwidth load.

CPU and bandwidth load from all switch ports could be collected with the help of SNMP traps. The SNMP traps should be stored in a database and the sampling process should start after a predefined threshold is reached, for example:

- A load of more than 90% for 30 minutes on a certain switch port;
- A constant data flow [1] of more than 4 Gb/s for one hour on a certain switch port;

To determine the best values for the threshold, the CPU and bandwidth load could to be measured for each switch port. To give an indication, how the results for this measurement will look like in a chart, we made an example in figure 4. It is now visible that if the load for a port, for example port 50, has a constant data flow above 4 Gb/s, the sampling process has to be started.

When two edge switches have this behavior, this *could* imply an exchange of data between the two edge switches. Starting the sampling process is required to determine the source and destination edge switch. As mentioned earlier, sFlow would be the best tool for collecting traffic flow information. Source and destination MAC addresses and/or source and destination switch ports must be extracted from the sFlow data.

Classification and prioritization of flows can be used to trigger a cut-through path. Flows with a short lifetime should not be able to trigger a cut-through path and should be forwarded on the default Internet VLAN. Constant flows over a longer period of time could trigger a cut-through path to optimize the traffic flows performance. A cut-through path should be triggered, when a large amount of traffic or a constant traffic flow is present in the following abstract scenarios, from:

1. a source customer router to a destination customer router (see figure 5(a));

[1] Constant data flow = Data keeps flowing with a minimum speed of 1 Gb/s. Constant flows will have no gaps in the flow.

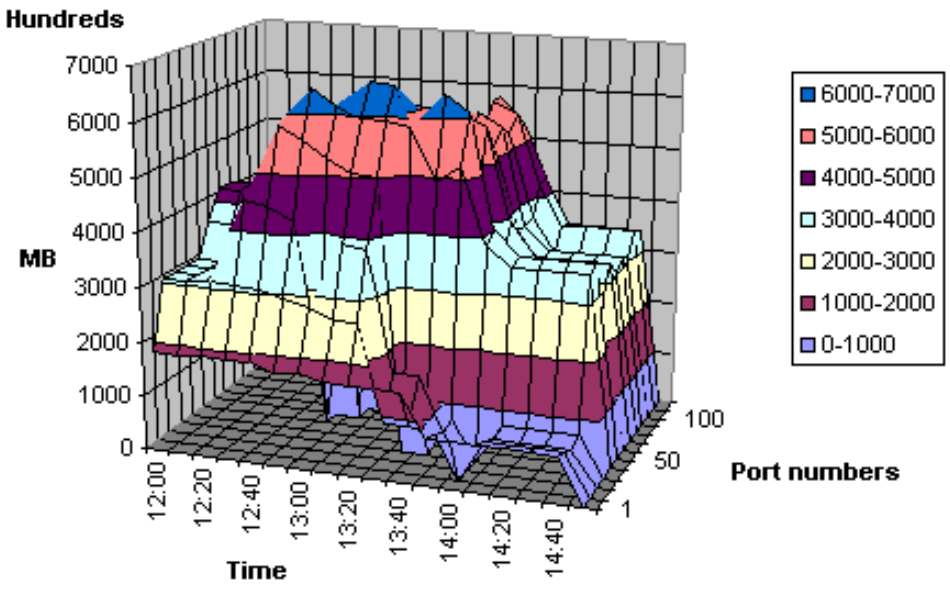


Figure 4: Bandwidth per switch port

2. different source customer routers, connected to the same edge switch, to one destination customer router (see figure 5(b));

3. different source customer routers, connected to the same edge switch, to different destination customer routers connected to the same edge switch (see figure 5(c));

4. a source customer router to different destination customer routers, connected to the same edge switch (see figure 5(d));

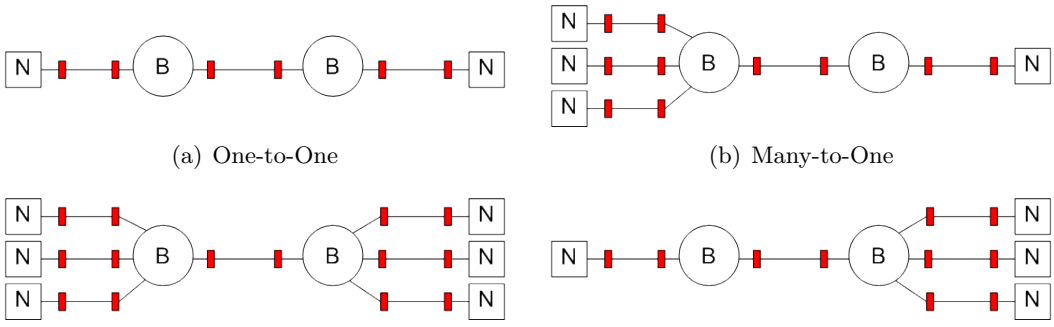


Figure 5: Abstract scenarios

The created cut-through path must be monitored and analyzed, because the cut-through path should be torn down when the traffic flow does not reach the threshold for a predefined time period. An overview of our sampling process can be found in figure 6.

This flow diagram shows that the first step is measuring the load on the different switches. sFlow sampling starts when the load/bandwidth comes above a predefined threshold. The samples are stored in a database, which can be used for the filtering process. Filtering per switch starts

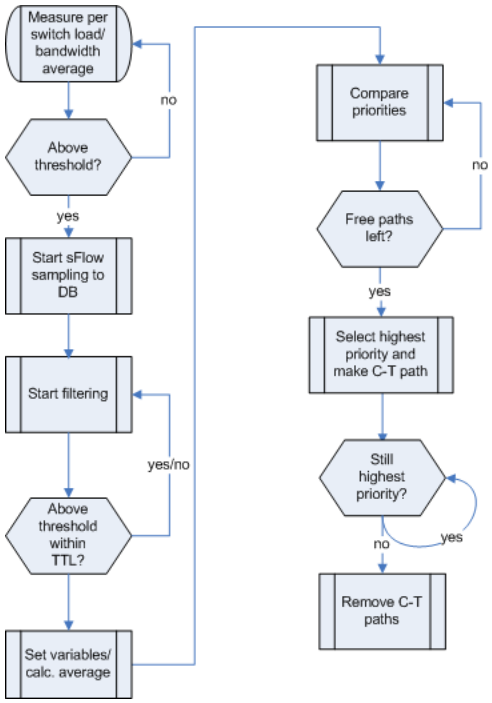


Figure 6: Flow diagram sampling process

when the sFlow data from a switch is collected. The filter record stored in the database can have the values shown in table 1.

SSwitch	DSwitch	VLAN	SPort	DPort	SMAC	DMAC	Count	Priority	STime	TTL

Table 1: Filter record

The Source Switch (SSwitch), Destination Switch (DSwitch), VLAN and Port fields are regular identifiers. To create a cut-through path for the scenarios in figure 5 the priorities in the records must reach a certain threshold before the Time To Live (TTL) is reached. If, for example, the TTL is 30 minutes and the threshold is reached or if a flow with a higher priority is detected within this time period, cut-through paths must be created for the flows with the highest priority.

The end time of the hold-down period can be calculated, by adding up TTL and Start Time (STime). The TTL value can be variable, the best starting value has to be determined after further research. The decision if cut-through paths have to be created or torn down is taken after each TTL period. The following sFlow sort processes on the control server deliver the priority values:

1. Sort flows per DSwitch, then per SPort & SSwitch and then per VLAN, the priority is set in the record according to how far the measured value is above the threshold for the minimum packet count;

2. Use the records from the first process, combine the total flows per SPorts from the SSwitch and count the total flows to a certain DSwitch. Calculate the sum of the priorities, plus the total packet count related to the total number of flows.

Process one covers the scenarios 5(a): *One-to-One* and 5(d): *One-to-Many*. Process two covers the scenarios 5(b): *Many-to-One* and 5(c): *Many-to-Many*. If the TTL for a record is reached,

the flows with high priorities get a longer TTL. Then the priority for that record is also based on the the average packet count in the TTL period.

For creating cut-through paths the control server determines by checking the available switch ports, if physical cut-through paths can be created. When enough resources are available, the control server automatically configures the cut-through path and adapts the (photonic) switch configuration if necessary. A specific flow will be forwarded over the cut-through path and the flow is further analyzed to determine the current priority. At the moment that the priority lowers and does not reach the threshold anymore, the cut-through path must be torn down and the traffic will be forwarded 'normally' through the core switch.

4.4.1 Bandwidth prediction

The AMS-IX network has a 24-hour traffic cycle which describes a cyclical behavior (see figure 7). Bandwidth prediction methods take advantage of the nature of traffic to determine the actual effective bandwidth required in the future.

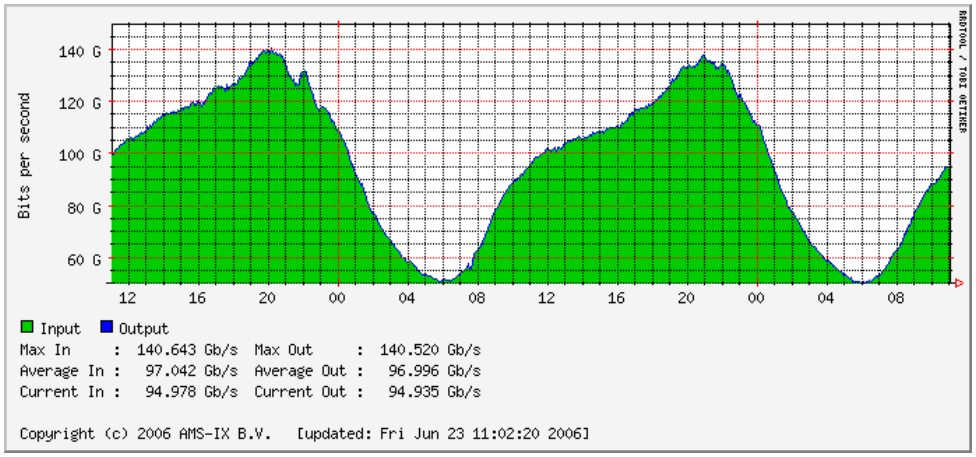


Figure 7: AMS-IX network statistics

Some research has been done in the area of bandwidth prediction. In each research project are different algorithms discussed. This leads to several solutions for bandwidth prediction methods. Balaji Krithikaivasan et al. [16] propose a dynamic bandwidth provisioning framework that can adapt to short-term traffic fluctuations. For their research they used real-life measured data. They prefer the ARCH [16] models over the ARIMA [17] model, because they think the ARCH model is more precise. For bandwidth prediction models for the cut-through solution within the AMS-IX, we refer to the following research projects [16], [17], [18]. The research projects give a good view into this subject.

The forecasts of traffic flows can be used to identify traffic flows with a long lifetime. These flows have to get a higher priority, so that the cut-through paths for these flows stays active for a longer period of time. This modification can lead to a more effective and intelligent approach for assigning the cut-through paths.

4.5 Cut-through creation

When creating the cut-through path, we have to determine which frames will be forwarded at the data layer. When the actual trigger has created a new path, the frames with destination

router(s) connected to the same egress switch will be forwarded over the new path. Each individual frame is forwarded based upon the state of the CAM table and MAC filters. With flow based forwarding all frames from the same flow will be forwarded along the same route. Frame forwarding is more efficient than flow forwarding, because other frames from other flows are also forwarded over the newly created path. This implies more bandwidth and less congestion at the core.

With Virtual LANs (VLANs) it is possible to connect multiple hosts from different LAN segments to a completely different logical LAN. This can be done based upon physical or logical addresses, protocol types, tagged frames or user defined rules. For example, different switch ports or logical addresses can be connected to one single broadcast domain.

The use of VLAN's could be a solution for the AMS-IX capacity problem. At the moment 99% of the traffic is part of the Internet VLAN. A cut-through path must be created at the moment that a certain traffic flow reaches the defined threshold. A VLAN should be created automatically for the specific traffic flow.

The edge switches need to configure the new VLAN and adapt the current trunk to the core switch. Traffic from the specific flows needs to be tagged with a new VLAN number. This can be accomplished with the help of filtering.

4.5.1 Filtering

Port assignment for the new VLAN on the edge switches is crucial to forward the new VLAN on the cut-through path. It is possible to filter frames and evaluate them and decide to which VLAN a frame belongs. As mentioned before, this can be based on switch port, MAC, IP address, high-layer protocols and on user defined rules.

With the latter it is possible to create VLANs based on a ruleset. For example, a VLAN can be created with the following rules:

```
All hosts with subnet address 192.170.10.x       (to be excluded)
IP addresses: 192.170.10.5, 192.170.10.6         (to be excluded)
MAC addresses: 03-3A-6A-01-9C-01-09
```

The 802.1Q standard [32] for VLANs specifies the *filtering database*. Information about VLANs is maintained in the filtering database. The filtering database stores the data for processing a frame. The database contains statically and dynamically gathered information. We need to control static filtering entries, because these entries can decide for each port, if frames are allowed to be send to a specific MAC address through the port. The static information entries are not automatically removed through aging, they can only be mutated through management. The filtering database also contains dynamic entries. The port, through which a frame with source MAC address and a specific VLAN ID (VID) arrives, is determined. These entries are learned by the bridge and cannot be created or mutated through management. The dynamic entries in the database are updated with MAC address and VID information.

The data frame processing at a bridge is done through the following steps. The first step is ingress rule checking. Where the received frame is checked against a set of rules, like if the VLAN tag is not allowed to have a value of 0xFFF and the frame types need to be valid. Also data frames need to be acceptable.

The second step is to check if the topology condition is active. This is done by checking the state of the bridge ports (forwarding or blocking state). The third step is applying the frame

filtering rules on the received data frame. This is done based on the VID and the destination MAC address, stored in the *filtering database*.

The following step is checking the egress rules. These rules check if the forwarding port is a member of the VLAN. Further is checked if the data frame must be tagged, untagged, encapsulated or decapsulated.

All frames are queued after the egress rules. The frames are chosen from the queue based on the priority associated with the frames. The last step before transmission is frame format modification and frame check sequence (FCS) calculations.

In the situation for the AMS-IX, it is not possible to make a cut-through decision to a newly created VLAN only on the source MAC or IP address. The customer routers can have only one MAC and IP address per router, but could also have two uplinks to an edge switch. Decisions made only on the source address, could forward traffic designated for other edge switches through the cut-through path. Decisions should be based on source port and destination MAC address defined in the static filtering entry.

This solution helps to cut off a specific traffic flow from the concentrated flow between an edge switch and a core switch, and forward this traffic on the new cut-through path. A difficult problem which remains, is the created loop.

4.5.2 MSTP

The loop can be solved with the help of Multiple Spanning Tree (IEEE standard 802.1S). MSTP quoted by IEEE [25]:

> " This Supplement to IEEE Std 802.1Q adds the facility for VLAN bridges to use multiple spanning trees, providing for traffic belonging to different VLANs to flow over potentially different paths within the virtual bridged LAN."

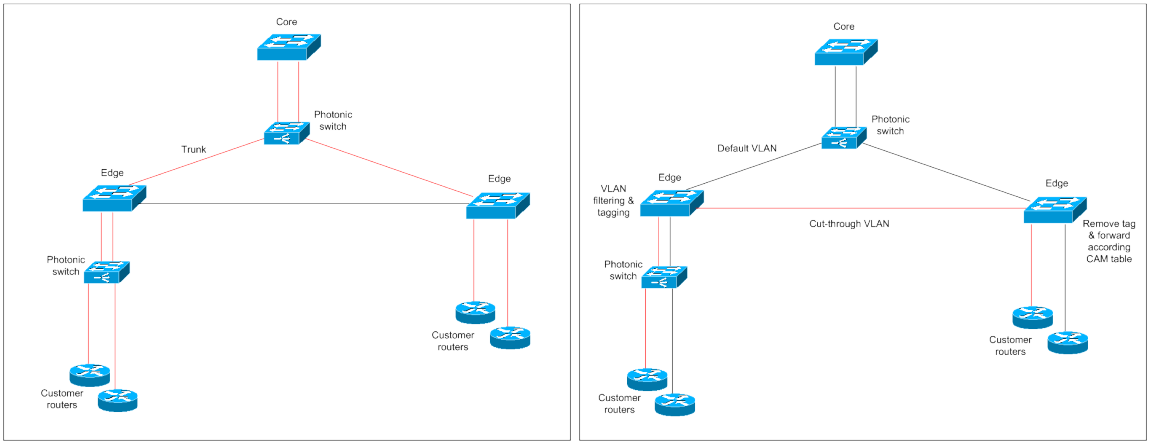
This means that a single STP instance per VLAN will be used. The STP computation can be influenced. This makes it possible to compute different STP instances per VLAN. The original VLAN (Internet VLAN) computes an STP instance, which blocks the cut-through ports on the edge switches (see figure 8(a)). The second VLAN (cut-through path) computes a STP instance, which blocks the port to the root bridge (see figure 8(b)).

This configuration prevents the creation of loops and makes it possible to use the cut-through path for a specific traffic flow. Issues concerning this solution are the automatic configuration of VLANs, trunks, filtering, port assignment, correct STP computation.

The multiple instances of STP are not necessary when the ingress and egress switch are configured correctly. Both switches should know which ports are allowed to forward traffic from the new VLAN. This will be the cut-through port and the port to the particular customer routers. In this situation the traffic will always be forwarded in the correct direction and loops will be prevented. The VLAN can be seen as a point-to-point connection. Another possibility is that the egress switch removes the VLAN tags and forwards the original frames through the port listed in the CAM table.

4.6 Cut-through path reassignment

To optimize the efficiency of cut-through paths, we periodically calculate the thresholds for the flows. We try to allocate network resources optimally, by keeping variables of the total free paths



(a) Default VLAN with STP instance

(b) Cut-through VLAN with STP instance

Figure 8: VLANs and STP

and occupied VLANs. The control server has to compare the total number of packets transmitted over the assigned cut-through paths with that in the newly calculated ones. Subtract the former from the latter. If the difference exceeds a threshold value, the control server has to reassign the cut-through paths.

The servers has to remove the old path and set up a new path. To set up a new path, it orders the photonic switch to setup a new path. Secondly, it orders the ingress switch to set up a new VLAN to the egress switch. The ingress switch sets up a new VLAN and the appropriate MAC filters after receiving a signal message from the control server. The egress switch also sets up the new VLAN. The mentioned steps in the process must repeated continually. With this operation we can efficiently manage the paths and VLANs in a non-stationary environment.

4.7 Draft

This paragraph describes a draft for cut-through paths. To implement this solution needs predefined thresholds, control management and controlled VLANs.

4.7.1 Threshold

Predefined thresholds are required to determine when traffic flows can trigger the creation of a cut-through path. AMS-IX needs to do a research on the topic of thresholds. It is very important to determine what the maximally allowed CPU and bandwidth load per switch should be. These thresholds are important for starting the sampling process. The different priority levels need to be determined too. Traffic flows with a huge amount of traffic, or constant flows which last for a long period of time, should obtain a high priority. The thresholds for the amount of traffic and the lifetime for a constant flow need to be set.

The sampling process helps to determine which edge switches and which customers are involved with a certain traffic flow. Parameters, like source and destination MAC addresses, need to be extracted from the samples. The parameters are required for the automatic creation of cut-through paths. Traffic which flows through a cut-through path, needs to be monitored and matched against the predefined thresholds. The cut-through path needs to be torn down

when the traffic does not reach the predefined trigger. The process of automatically creating cut-through paths dependents on the correct determination of the needed thresholds.

4.7.2 Control management

The implementation of cut-through paths within the current AMS-IX network requires excessive management. A Control Server architecture is the preferred method for creating, managing and tearing down automatic cut-through paths. The architecture will manage the network centrally and should be build redundantly. The control architecture should be a dedicated management network to separate 'production' traffic from the management traffic to prevent interference with each other. The filtering and configuring processes, within the control architecture, are very important within the whole process.

The filtering process covers the sampling process, data computation, matching algorithms and parameter extraction. These processes require a lot of system resources. This requires strong high-end dedicated systems, which need to be secured against attacks. The segmentation of production and management network improves the management networks security. The control architecture should be a secure private network. The control process must also be physically separated from the filtering process, because the filtering process may not disturb the control process. The control server should be build redundantly to enable the possibility of smooth failover in the event of a system failure. It is also very important to securely validate all configuration steps. This is done to prevent malicious configurations to be executed.

The configuring process covers the physical and logical creation of cut-through paths in the network. After the filtering process, the control architecture needs to automatically configure all the required components for the cut-through path. The needed parameters are supplied by the filtering process. An application needs to be developed which enables the control server to execute the configurations on the network components. SNMP or scripts which are executed over SSH/SSL tunnels could be an option for the configuration.

4.7.3 VLANs

When this configuration is possible, the use of VLANs in association with the control architecture should be a possible solution for now. Currently AMS-IX uses different VLANs, but 99% of the traffic is tagged with the Internet VLAN. Traffic, which travels through a created cut-through path, must be tagged with a new VLAN number. A new VLAN needs to be configured for a cut-through path. Filters are needed to filter the traffic, which is allowed to travel through the cut-through path.

Figure 9 is used to explain the possible VLAN configuration. The figure shows four different client routers (R1 to R4), two edge switches (Edge1 and Edge2), different port names (p1 to p8), two trunk links (TRUNK1 and TRUNK2) and a core switch.

A flow with a huge amount of traffic is flowing from R1 to routers R3 and R4 (scenario 5(d): *One-to-Many*). Normally the traffic is tagged with the Internet VLAN (VLAN 501) at the ingress switch. The flow travels from the ingress switch through the core to the egress switch and from there to routers R3 and R4. The big flow will trigger the creation of a cut-through path.

The first step is to create a new VLAN number (VLAN 520) on both edge switches. The photonic switch connects p4 and p5 physically and the new VLAN is assigned to the appropriate ports (p4 and p5). Ports p4 and p5 are in blocking mode and will not forward or listen to any traffic.

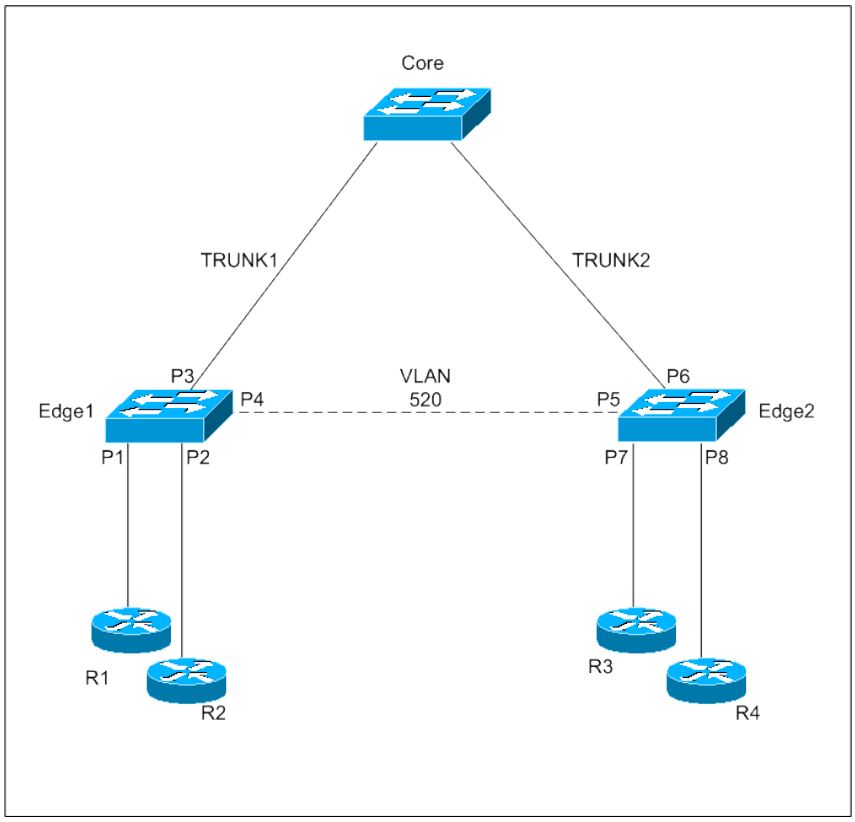


Figure 9: VLAN example configuration

The next step is creating a MAC filter based on R3's and R4's destination MAC addresses. The MAC addresses are obtained from the control server. An possible MAC filter is shown below.

```
MAC-Filter CutThroughMAC:
       MAC R3
       MAC R4
```

The MAC filter needs to be applied to two different frame filters. One filter for VLAN 501 and one for VLAN 520. Possible implementations of both frame filters are shown below.

```
Frame-Filter 520
       match MAC-addresses CutThroughMAC
       action allow forwarding on p4
       action deny anything else on p4

Frame-Filter 501
       match MAC-addresses CutThroughMAC
       action deny forwarding on p3
       action allow anything else on p3
```

The last step is configuring an egress filter on switch port p4. This egress filter encapsulates the Internet VLAN tagged frames (VLAN 501) with the new VLAN tag (VLAN 520). This

encapsulation can be compared with the 802.1ad (Provider Bridges) standard [19]. 802.1ad is developed to allow service providers to impose their own VLAN partitioning without disturbing customer VLAN partitioning. 802.1ad encapsulated customer VLANs with the provider VLANs. Ports p4 and p5 should switch to forwarding mode directly after applying the filters. This prevents frames from being discarded.

The egress switch needs an egress filter to decapsulate the VLAN 520 tagged frames received on port p5. After decapsulation the frames look exactly the same as previously received frames on p6. The egress switch only needs to forward the frames to the correct egress switch port.

This is a solution for unidirectional flows. When flows are bidirectional, the same procedure needs to be followed in the opposite direction. The draft can be used for all scenarios from figure 5. The only difference will be the MAC filter. The filter will contain one MAC address for *One-to-One* and *Many-to-One* scenarios. The filter will contain more than one MAC address for *One-to-Many* and *Many-to-Many* scenarios.

5 Hybrid bridges

This chapter describes hybrid bridges, which are related to the creation and management of cut-through paths in a layer two network, like AMS-IX. This technology is still in childhood, but represent the ideal solutions with regards to the AMS-IX network and the creation of cut-through paths in a layer two network.

5.1 TRILL

Radia Perlman [20] and Joe Touch [21] are part the TRILL Working Group, which does research on the Transparent Interconnection of Lots of Links (TRILL) problem. TRILL focuses on multipath switching in a single spanning tree. The most recent draft [22] dates from June 9th 2006.

TRILL discusses the problems and required properties of solutions with regard to multipath switching in a single spanning tree.

5.1.1 Problems

Chapter 2: Problem definition describes problems related to the automatic creation of cut-through paths in the AMS-IX network. Problems addressed to layer two networks, which are discussed in TRILL, are:

1. *Inefficient paths*
   With STP some flows will not use the most optimal path in the network and some links will concentrate different flows into one link. This means that different flows have to share the bandwidth of that particular link;

2. *Convergence*
   Small changes in the topology causes STP to compute a new spanning tree. The convergence of the topology could take some time, because of the propagation and distribution of BPDU's (Bridge Protocol Data Units);

3. *Backup paths*
   Layer two networks do not have the ability for using alternate, backup paths. This in comparison to layer three routing, were alternate, backup paths are used for load balancing and rapid convergence;

4. *Ethernet extensions*
   There are more protocols besides STP, like RapidSTP (802.1W [23]), VLANs (802.1Q [24]) and MultipleSTP (802.1S [25]), which complicate the configuration. Furthermore are these extensions not invented for multipath routing in a single spanning tree;

Most of the problems discussed in the TRILL draft are strongly related to the problems with regards to the AMS-IX network. Given this information, the proposed TRILL solutions could represent possible solutions for the automatic creation of cut-through paths in the AMS-IX network.

5.1.2 Solutions

The TRILL Internet draft [22] describes some desired properties of solutions to solve the TRILL problems. Some of these solutions are:

1. *Services*
   TRILL solutions should not take away any of the services currently provided by layer two networks, like unicast, broadcast and multicast;
2. *Added services*
   Use of multipath switching in a single spanning tree;
3. *Forwarding Loop Mitigation*
   A TTL (Time To Live) field should be introduced to limit the impact of loops or detect them explicitly;
4. *VLAN*
   Multiple VLAN extensions should be supported by the TRILL solution, like 802.1Q [24], 802.1V [26] and 802.1S [25];
5. *Security*
   New security vulnerabilities should not be introduced when implementing TRILL solutions. TRILL solutions should be vulnerable to the same or to less security issues in comparison to current layer two technologies;

At the end TRILL solutions are intended to address problems of path efficiency and stability within a layer two network. TRILL solutions should take as less configuration as possible and should not interrupt current layer two services and technologies.

5.2 RBridges

Radia Perlman and Joe Touch are also part of a Working Group working on the RBridge Protocol. The RBridge Protocol and the RBridge Architecture are described in two different Internet drafts [28] [29]. This paragraph is a summary of the mentioned Internet drafts [28] [29] and tries to explain the protocol and the architecture of RBridges with regards to the TRILL and AMS-IX problem.

RBridges (routing bridges) combine the advantages of bridges and routers. RBridges, just like bridges, are transparent to IP nodes and solve the problem of loops, like routers do. A RBridge must support the same technologies as a regular bridge does, like VLAN support.

5.2.1 Protocol

RBridges use Ethernet headers, just like regular bridges, but for RBridges to work properly, the current Ethernet header [30] needs to change. The current Ethernet header is displayed in table 2.

Bytes	7	1	6	6	2	46 - 1500		4
Definition	Preamble	SFD	DestAddr	SrcAddr	Type	PDU	PAD	FCS

Table 2: Ethernet header

Instead of changing the current Ethernet header standard, a RBridge will encapsulate an Ethernet frame in a understandable frame for both RBridges and regular bridges. The encapsulated frames travel between RBridges and regular bridges. This frame encapsulation is done:

- to add a hop count to frames in transit;
- to prevent the learning of source MAC addresses from frames in transit;
- to add RBridge addresses for frame routing purposes;

The encapsulated headers should look like a regular Ethernet header to Ethernet bridges, which can be situated between two RBridges. The difficulty is that the header should look like a regular Ethernet header, but the RBridge should be able to determine that the frame has an encapsulated header and is not a regular Ethernet frame. Therefore, the Working Group proposed a new Ethertype with the name "RBridge encapsulated frame". The new Ethertype helps RBridges to distinguish regular Ethernet frames from RBridge encapsulated frames.

An RBridge encapsulated frame looks like the frame in table 3:

outer header			**shim header**		**original frame**
DestAddr RBr.	SrcAddr RBr.	Prot. type	TTL	egress/ingress RBr.	original frame

Table 3: RBridge encapsulated frame

Besides the original header the encapsulated frame contains an outer header and a shim header. The different fields in the outer and shim header have the following functionality:

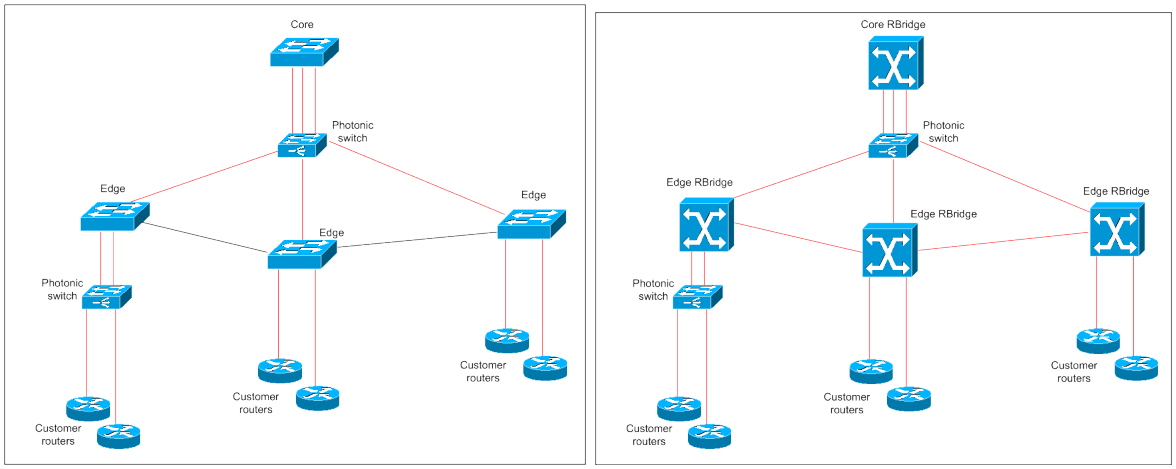
- **Outer header**
  - *DestAddr RBr.*
    Layer two destination of the next RBridge in the case of a unicast frame. For flooded/broadcast frames this address will be a new (to be assigned) multicast layer two address meaning "all RBridges";
  - *SrcAddr RBr.*
    Layer two address of most recently transmitting RBridge;
  - *Prot. type*
    Protocol type, which identifies the encapsulated RBridge frame. This protocol type still needs to be assigned. The proposed protocol type is "RBridge encapsulated frame";
- **Shim header**
  - *TTL*
    Time-to-Live value which will be decremented by each RBridge. If the TTL reaches a value of 0 the frame will be discarded;
  - *egress/ingress RBr.*
    In the case of a unicast frame this field contains the layer two address of the egress RBridge. The field contains the layer two address of the ingress RBridge in the case of a multicast frame;

RBridges have to use some type of routing protocol as a control plane to make switching decisions and to take advantage of switching loops. The routing protocol chosen is a link state routing protocol, which has the characteristics of IS-IS [31]. A RBridge can be compared with a router running a level 1 routing protocol in an area. A RBridge is able to forward frames with a known unicast destination, frames with a unknown unicast destination, layer two broadcast frames and layer two multicast frames in a layer two network with loops.

5.2.2 Architecture

The architecture and working of RBridges will be explained with the help of a topology similar to the AMS-IX topology. The topology can be seen in figure 10(a). With the creation of a cut-through path with regular bridges, a link between two edge switches, a loop will be created in the network. STP takes care of this loop and computes a single spanning tree. The red links indicate the active links after STP computation.

STP blocks the cut-through path and all the traffic is still concentrated on specific links. Links are still not used efficiently. If RBridges replace regular Ethernet bridges or if RBridges are implemented amongst regular Ethernet bridges, alternate/backup links will be used more efficiently. The regular Ethernet bridges are replaced by RBridges in figure 10(b). All the links, including the cut-through paths, can be used for switching Ethernet frames more efficiently.



(a) Available paths regular bridges

(b) Available paths RBridges

Figure 10: Available paths

RBridges make it possible to implement a full mesh in a layer two network. RBridges also have the ability to segment or partition a layer two network with regards to STP computation with regular Ethernet bridges. Only regular bridges between RBridges will compute a single STP. RBridges have the ability to not, just like routers, partition in the STP computation. This is possible, because RBridges can interact with received BPDU's from regular Ethernet bridges using one of the following interaction models:

- *Transparent Participation*
  This type of participation enables the RBridge to broadcast BPDU's on all other links except the receiving link. The RBridge campus would appear as a transparent device

(like an hub) on the network. This means that RBridges will not participate in the STP computation;

- *Active Participation*
  Active participation means that the RBridge actively participates in the STP computation. A RBridge may receive BPDU's and may emit new BPDU's on the network. In this model a RBridge emulates a 802.1D bridge. This method is not recommended, because RBridges use a RBridge routing protocol instead of an STP instance to compute the most efficient paths in the network;

- *Blocking Participation*
  This model allows a RBridge to partition the total network spanning tree into smaller spanning trees. To do this the RBridge completely blocks received BPDU's;

Before RBridges work properly, some type of peer and topology discovery has to be accomplished by the RBridges. This discovery should be done in the same way the chosen RBridge link-state routing protocol would function. The topology discovery is important to make efficient use of links and switch traffic in the most cost-effective way. The protocol messages needs to be distinguished by RBridges from regular Ethernet frames. This can be accomplished by using a different protocol type.

5.2.3 Rbridge Operation

Every RBridge has to go through some operational phases before working properly. Dependent of the RBridge function (e.g. ingress or egress RBridge) the RBridge has different phases. Some operational phases need to be fulfilled by all RBridges. These operations are:

- *Peer and topology discovery*
  As state earlier, this phase covers the discovery of other RBridges in the network. Every RBridge builds a topology table to determine the most efficient paths to another RBridge;

- *Designated RBridge election*
  When a node (e.g. host, hub, bridge or router) can connect to two different RBridges, a designated RBridge has to be elected. The chosen designated RBridge takes care of the forwarding, encapsulation and decapsulation of frames for the specific Ethernet segment where it is dedicated for;

- *Ingress RBridge Tree computation*
  Tree generated by each RBridge for delivery of broadcast, multicast and flooded frames to all other RBridges;

- *Link-state routing*
  This is the learning and forwarding state. The RBridge starts learning MAC addresses from received frames and forwards unknown destinations according to the local Ingress RBridge Tree;

- *Advertisements*
  The RBridge periodically propagates its known destinations to other RBridges in the network with the help of advertisements;

Ingress and egress switches have some additional phases. These additional phases, besides RBridge and MAC learning, forwarding and advertisements, are:

- *Encapsulation*
  An ingress RBridge forwards an ouside frame onto the RBridge network. Ingress RBridges encapsulate a frame with the outer and shim header. These headers determine the egress RBridge, TTL and additionally the next hop RBridge;

- *Decapsulation*
  An egress RBridge decapsulates a received frame and forwards the frame via the egress interface onto the Ethernet segment. The egress RBridge is the exit point from the RBridge network to the regular network;

There is another type of RBridge. This RBridge provides transit in regard to frames. Two possible models for transit forwarding RBridges are edge-to-edge and hop-by-hop. The determination of the encapsulation is the main difference between the two models. Four types of traffic can be distinguished and forwarded. These types are:

- *Unicast frame*
  The main difference between edge-to-edge and hop-by-hop forwarding of a unicast frame is the information in the outer and shim header. If a frame is forwarded edge-by-edge the destination address in the outer header and the destination address in the shim header are the same and remain the same during transportation of the frame. If a frame is forwarded hop-by-hop, the destination address in the shim header is equivalent to the address of the egress RBridge switch. This address remains the same during transportation through the network. The destination address in the outer header specifies the address of the next hop RBridge. This address changes per hop. The next hop RBridge is determined after examination of the local forwarding table. The source address in the outer header also changes to the address of the most recently forwarding RBridge;

- *Non-unicast broadcast frame*
  Broadcast frames are forwarded according tot the Ingress RBridge Tree (IRT). The IRT contains a tree with paths going toward all RBridges. Broadcast frames are encapsulated and forwarded on interfaces which connect to peer RBridges and broadcast frames are decapsulated and forwarded on egress interfaces. The interface type can be determined from the IRT;

- *Non-unicast multicast frame*
  Forwarding of multicast frames works similar to the forwarding of broadcast frames. A group multicast MAC destination address is used to forward the frame according to a RBridge's IRT. A RBridge recognizes a multicast frame by the group multicast MAC address. This knowledge enables a RBridge to prune a multicast frame if no multicast participants for the particular group are connected to the RBridge;

- *Non-unicast frame flooding*
  Frame flooding can also be compared with the forwarding of broadcast frames, but the flooding mechanism could be optimized. An ingress RBridge will encapsulate a frame with an unknown destination with a MAC destination broadcast address. The encapsulated frame is forwarded according the Ingress RBridge Tree. All transit RBridges could decide to forward the frame according their local IRT to all egress RBridges. In this situation the transit RBridge does not analyze the original header. An optimization would be if the transit RBridge decapsulates the flooded frame and checks if it knows the destination. If the transit RBridge has a known destination, the RBridge does not forward the flooded

frame anymore, but sends a reply to the ingress RBridge. This prevents the frame from being flooded through the whole RBridge network;

5.3 Draft

In the future cut-through paths will probably be obsoleted by technology like RBridges. RBridges provide better layer two forwarding than current 802.1D bridges. A RBridge is a hybrid router/bridge designed to provide layer two connectivity. RBridges use a routing algorithm to automatically route layer two traffic. All RBridges build a topology table to have knowledge of the complete layer two network. Forwarding decisions are made with the help of metrics (routing protocol dependent) and the topology table. A RBridge encapsulates and decapsulates a regular Ethernet frame for making a clear distinction between the two different frames. The encapsulated frames have two additional headers, which provide extra information like source and destination RBridge address and TTL.

A RBridge supports mechanisms for handling unicast, broadcast, multicast en flooded frames to known and unknown destinations. Loops are allowed in the layer two network, without the danger of creating broadcast storms in the present of broadcast traffic. This because of the added TTL field, which is decremented per hop and frames with a TTL of 0 will be discarded. RBridges use redundant links for automatic failover in the event of a link failure.

RBridges make it possible to change the hub & spoke topology in a full mesh topology. A full mesh topology provides more alternative links in the event of a link failure. The routing algorithm determines the best path for a specific traffic flow. The traffic will therefore always travel on the most efficient way. Multicast frames can be pruned by a RBridge to drop packets on interfaces where no multicast listeners are present. This prevents unnecessary forwarding of multicast frames.

The AMS-IX can also keep using the current redundant hub & spoke topology. When all Ethernet bridges are replaced by RBridges the Virtual Switch Redundancy Protocol is not needed anymore. The routing algorithm determines the most efficient and cost-efficient path from edge RBridge to edge RBridge.

For now it is not clear whether RBridges will be a dedicated hardware solution or a firmware solution, which can be implemented in current switches. The latter would be the most ideal solution for AMS-IX, because the current switches do not need to be replaced.

6 Photonic switch

The currently used photonic switches could also provide a solution for AMS-IX. All customers with 10GE connections are connected to a photonic switch and the photonic switch connects the users to an edge switch. The photonic switch can be compared with an intelligent patch panel.

6.1 MOEMS

Photonic switches include MEMS [33] devices and optical components. Devices including MEMS and optical components are called Micro-Opto-Electro Systems (MOEMS). Photonic switches are not tied to specific data rates and protocols. Photonic switches direct the incoming bits to the egress port, no matter what the line speed and protocol are. Optical switches have the ability to separate signals at different wavelengths and direct them to different ports. Figure 11 shows tiny mirrors, which reflect the input bits to the correct egress port.

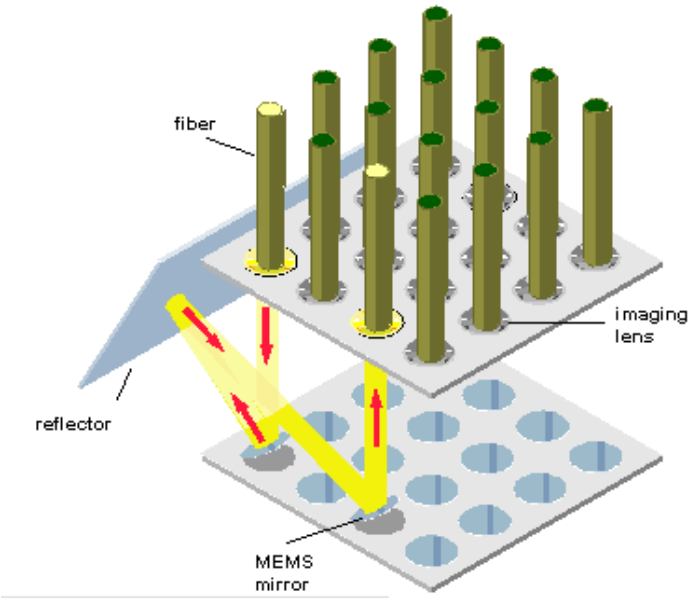


Figure 11: Photonic switch

By changing the mirrors angle, the input bits could be redirected to another output port. Another customer could be connected to this output port. The photonic switch has created a new path from the same source to another destination. If this mechanism could be managed automatically, the photonic switch could be used for the creation of automatic cut-through paths.

6.2 Two uplinks

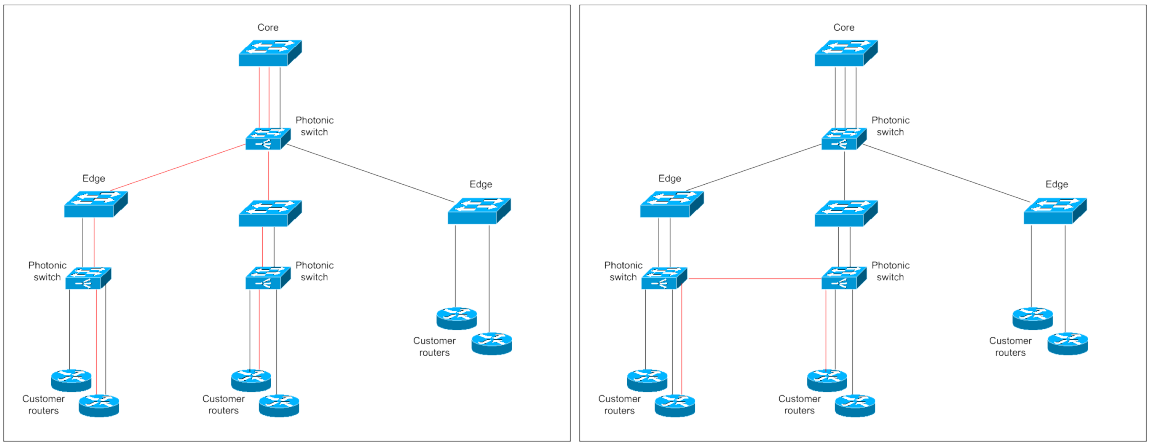
To implement a photonic switch as a solution, AMS-IX should be able to automatically configure the customer router or the customer should manually configure his router. This latter method is not preferred, because this needs interference from the customer. A customer needs at least two uplink interfaces to the AMS-IX network in this solution. One for the default traffic and one interface for the cut-through traffic. The actual configuration involves changing the routers routing table.

The photonic switch functions as an intelligent patch panel. This means that if a traffic flow between two customer router reaches a certain threshold, the customer routers could be directly connected with each other. To do this, the photonic switch needs to dynamically create a connection between the two customer routers. The difficulty here are the customer routers. The routers should be configured to send the specific traffic flow through the cut-through interface.

When the threshold is reached, certain parameters are supplied by the control architecture, like source and destination MAC address. The source and destination MAC address have to be looked up in a table to determine the ingress and egress switch and the associated switch ports. This information should be the basis for automatically creating a cut-through path in the photonic switch. The customer router should use the parameters to filter the traffic and send all the traffic which applies to the parameters through the second cut-through link.

This solution does not only lessen the load on the core switch, but also lessens the load on the edge switches, because the traffic flow will not travel through the edge switches anymore. Instead the traffic flow only travels through one or more photonic switches.

Figure 12(a) shows the regular traffic stream with the red links. Figure 12(b) shows the established cut-through path between the photonic switches. In this situation the traffic does not flow through the core and edge switches. If the customer routers are connected to the same photonic switch, the cut-through path will be established in that specific photonic switch. This represents a solution for point-to-point connections. The connection can also be made between a photonic switch and a destination edge switch. The edge switch will receive the cut-through traffic and looks up the destination MAC address in the CAM table. Next the switch forwards the frames through the appropriate egress switch port. This approach represents a solution for point-to-multipoint connections.



(a) Regular traffic stream

(b) Photonic switch cut-through stream

Figure 12: Photonic switch traffic streams

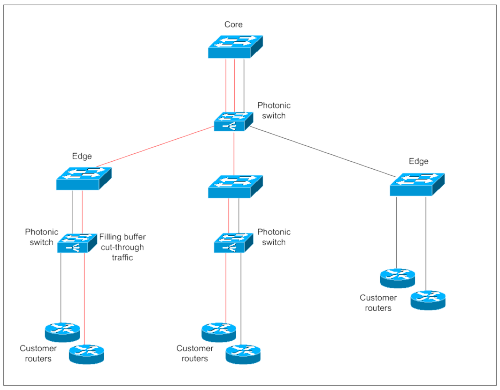
6.3 Burst switching

Another solution could be using a photonic switch with large buffer and filtering capabilities. The photonic switch collects a burst of traffic from a specific source to a specific destination. The source and destination are determined again with the sampling process and associated parameters mentioned in chapter 4. If a certain threshold is reached, the photonic switch starts

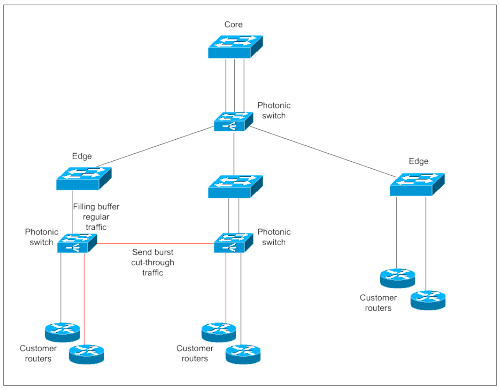
buffering traffic from the specific traffic flow. This solution could be compared to Optical Burst Switching [34].

To collect a burst of traffic, the photonic switch needs to filter all the traffic corresponding the traffic flows of the reached threshold and stores this traffic in a buffer. This filtering could be done on the destination MAC address of the customer router or on an destination IP prefix. When the burst of traffic reaches another threshold (e.g. maximum buffer size), the current connection from the photonic switch to the edge switch is torn down and a new connection is established between the photonic switch and the destination edge switch. After establishing the path the photonic switch sends a burst of traffic to the destination edge switch and tears the connection down after sending the burst. In the meantime of sending the burst, the regular traffic needs to be buffered by the photonic switch. After the burst is send and the cut-through path is torn down, the original connection is established again and the regular traffic will be forwarded. If the traffic flow still reaches the threshold, another burst of traffic could be collected.

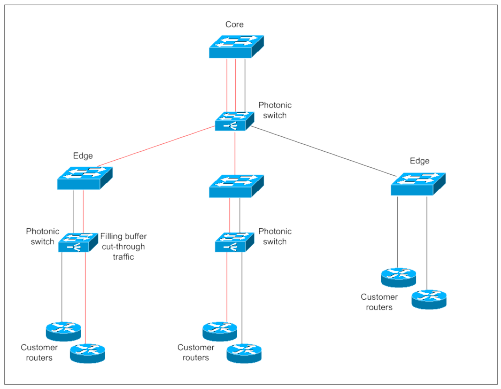
This solution requires photonic switches with a large buffer and intelligent filtering capabilities, but could be successful if such photonic switches would exist. The total process is reflected in figures 13(a), 13(b) and 13(c). Latency will be a disadvantage of this solution. All traffic, which is placed in the buffer, will have a higher latency in comparison to regular traffic forwarding.



(a) Filling buffer



(b) Send burst



(c) Filling buffer again

Figure 13: Burst switching

Burst switching could be implemented between two photonic switches, but also between a photonic switch and an edge switch or directly between two edge switches. In all solutions, the used

network components should be able to filter traffic and store traffic in a buffer.

6.4 Draft

Both solutions, two uplinks and burst switching, are not ideal solutions, but possibilities. The first solution requires configuration on the customer router. This can be done by AMS-IX or by the customer. It would be preferred if AMS-IX had the opportunity to configure the customer router, but this will hardly happen. This means that AMS-IX will have a lot more network components to manage and the routers are still customers property. Configuration by the customer is not preferred, because it could take some time before the customer configures his router to forward the correct traffic over the cut-through path.

Burst switching requires a very intelligent (photonic) switch, which has the capability to buffer a large amount of traffic. The biggest disadvantage presented by this solution is latency. Frames, stored in a buffer, could have a large latency. Large latencies can have an enormous impact on business-critical and time-sensitive traffic. A lot of customers complaining about large latencies will have a negative impact on the market position of AMS-IX.

7 Security

Security is a hot item nowadays. More and more organizations use all sorts of solutions to implement counter measurements to prevent attacks on the organizations network. Threat and anomaly detection is possible with sFlow data analysis, supported by the AMS-IX Foundry products. The AMS-IX is one of the biggest Internet Exchanges in the world. This makes it possible to detect worm propagation attacks, virus attacks and/or DDoS attacks on large scale. AMS-IX can use this information to inform their customers about possible attacks, so the customers can take counter measurements to prevent these attacks.

The main item for creating a cut-through path is the amount of traffic between two edge switches. A cut-through path should be established after the amount of traffic between to edge switches reaches a certain threshold. When this happens, a lot of traffic is traveling between two edge switches or between two customer routers. This amount of traffic could originate from multicast traffic or from connections to the available content provided by one of the AMS-IX customers. The amount of traffic could also originate from another kind of traffic, like traffic associated with a Distributed Denial Of Service (DDOS) attack. The information out of the filtering process could help by the detection of threats. Does the AMS-IX wants to create a cut-through path for a DDOS attack or is this not a problem and does AMS-IX only forward traffic without the concern of possible attacks?

The data analysis costs time and money, but if there is no policy to forbid traffic analysis, the AMS-IX should consider the possibilities of traffic analysis.

8 Conclusion

AMS-IX will have a capacity problem in the near future (approximately in one and a half year). AMS-IX started seeking for a solution to the capacity problem, but what would be the best solution? This question is hard to answer. Currently AMS-IX uses 10 Gb/s switch ports, which are configured in a channel. These channel configurations have restrictions with regards to scalability. The design of 100 Gb/s capable switch ports would be a solution for another undefined period of time.

RBridges will be the most ideal situation to switch traffic efficiently. RBridges are still in development, but allow the AMS-IX to create a full mesh layer two topology, which uses all the available paths in the most efficient way. Protocols like STP and VSRP are not needed anymore with this configuration. Implementation and test environments are not realized yet. It is not clear when RBridges are ready for implementation, but the development time will approximately be one to two years. It is also not clear yet, if RBridges become a dedicated hardware or a firmware solution.

An interim solution could be the use of VLANs. These VLANs have to be configured automatically. Cut-through paths should only be created if a specific traffic flow reaches a certain threshold. These thresholds need to be defined. A control architecture takes care of the sampling, filtering and computation process. The control architecture triggers the network components to adjust their configuration for the creation of cut-through paths.

This paper describes some proposed drafts. This paper is based on theoretical research. The proposed drafts, like paragraph 4.7, are not tested in a physical environment, but based on theoretic research and assumptions. The realization of test environments and practical research was not possible due to a short time period of four weeks.

AMS-IX has the network components and tools to test the proposed drafts. AMS-IX needs to consider their current possibilities. They can keep using the current network without applying any changes, which means that AMS-IX will have serious capacity problems in the near future or wait for the development of RBridges and 100Gb/s capable switch ports. The second possibility would be doing research on the proposed control architecture and VLAN implementation. This could provide a possible solution for a longer time period.

9 Future

Implementation of the proposed drafts needs further research. Is the proposed interim solution feasible and affordable for AMS-IX? This question is very important when considering possible implementation. The required thresholds for different steps of the control architecture need to be defined. Research is required to determine the right values for these thresholds.

The different Ethernet switches and the photonic switches need to be configured automatically. Are there any standardized protocols which can be used for this process? SNMP could maybe be a possible solution, but is all configuration possible with SNMP?

The sampling process will store data samples in a database. Specific parameters from this database are needed for the configuration of cut-through paths. The database type is important to speed up the filtering and computation process. Research on different database types is recommended to choose and implement the most efficient database type for the control architecture.

Sampling traffic with sFlow and creating cut-through paths for flows with a high volume rate or constant flows could help adding additional security measurements to the AMS-IX network. Further research is recommended to determine the possibilities of anomaly detection with sFlow.

Research on the control architecture software is recommended. There are already different (layer three) software packages developed for controlling networks and network components. These software packages are not fully compliant to the proposed control architecture. Software packages need to be adjusted or a new package needs to be developed, which is fully compliant to the proposed control architecture. Point of interests during this development are memory usage and CPU performance. The sampling, computation and filtering process needs to be fully optimized to achieve the most optimal performance.

Technics, like GMPLS, are used for a dedicated end-to-end connection and could maybe foresee a possible solution. We have not done thorough research on GMPLS, because our time was short. Currently GMPLS is possible for point-to-point connections and not for point-to-multipoint or multipoint-to-multipoint connections. Chris Tracy, working on the GMPLS DRAGON project, informed us about an IETF draft [35], which covers point-to-multipoint GMPLS. The network components used by AMS-IX need to comply to RFC 3945 [36]. The AMS-IX network components do not comply to the RFC according to the Foundry Networks component specifications. Appendix C: GMPLS Architecture displays a possible GMPLS architecture within the AMS-IX network.

References

[1] Wikipedia about STP,
http://en.wikipedia.org/wiki/Spanning_Tree_Protocol

[2] Amsterdam Internet Exchange,
http://www.ams-ix.net/

[3] AMS-IX Traffic Statistics,
http://www.ams-ix.net/technical/stats/

[4] Virtual Switch Redundancy Protocol,
http://www.foundrynet.com/services/documentation/bigiron_rx_config/vsrp.html

[5] Foundry Networks,
http://www.foundrynet.com/

[6] Glimmerglass,
http://www.glimmerglass.com/

[7] Quarantine VLANs at AMS-IX,
http://www.ams-ix.net/technical/qvlan.html

[8] Allowed Traffic Types on Unicast Peering LANs,
http://www.ams-ix.net/technical/allowed.html

[9] SFlow RFC 3176
http://www.faqs.org/rfcs/rfc3176.html

[10] Building Business Intelligence from the Network
http://www.foundrynet.com/solutions/appNotes/BusinessIntelligence.html

[11] N. Duffield, C. Lund, and M. Thorup. Charging from sampled network usage. In SIGCOMM Internet Measurement Workshop, Nov. 2001.
http://public.research.att.com/~duffield/papers/DLT01-usage.pdf

[12] Packet Sampling Basics
http://www.sflow.org/packetSamplingBasics/index.htm

[13] Wikipedia about Simple Network Management Protocol
http://en.wikipedia.org/wiki/Simple_network_management_protocol

[14] Elisa Jasinska Homepage
http://www.jasinska.de/

[15] Cut-through Optical Path Control Technology for a Terabit-class Super-network, Kenichi Matsui, Takeshi Yagi, Yuuichi Naruse, and Junichi Murayama, 2004

[16] ARCH-based Traffic Forecasting and Dynamic Bandwidth Provisioning for Periodically Measured Nonstationary Traffic, Balaji Krithikaivasan, Yong Zeng, Kaushik Deka, and Deep Medhi
http://www.sce.umkc.edu/~dmedhi/papers/kzdm-ton-06.pdf

[17] Adaptive Bandwidth Provisioning Envelope based on Discrete Temporal Network Measurements, Balaji Krithikaivasan, Kaushik Deka1 and Deep Medhi, School of Computing and Engineering, University of Missouri-Kansas City, Kansas City, MO-64110, USA
http://www.ieee-infocom.org/2004/Papers/37_3.PDF

[18] Bandwidth-demand prediction in virtual path in ATM networks using genetic algorithms, N. Swaminathana, J. Srinivasanb, S.V. Raghavana
http://www.cs.ucsb.edu/~almeroth/classes/F99.595N/1581.pdf

[19] 802.1ad - Provider Bridges
http://www.ieee802.org/1/pages/802.1ad.html

[20] Radia Perlman
http://research.sun.com/people/mybio.php?uid=28941

[21] Joe Touch
http://www.isi.edu/touch/

[22] Transparent Interconnection of Lots of Links (TRILL): Problem and Applicability Statement, J. Touch and R. Perlman, 2006
http://ietf.cnri.reston.va.us/internet-drafts/draft-ietf-trill-prob-00.txt

[23] 802.1W - Rapid Reconfiguration of Spanning Tree, IEEE
http://www.ieee802.org/1/pages/802.1w.html

[24] 802.1Q - Virtual LANs, IEEE
http://www.ieee802.org/1/pages/802.1Q.html

[25] 802.1S - Multiple Spanning Trees, IEEE
http://www.ieee802.org/1/pages/802.1s.html

[26] 802.1V - VLAN Classification by Protocol and Port, IEEE
http://www.ieee802.org/1/pages/802.1v.html

[27] Cisco Networking Academy Program: Second-Year Companion Guide, second edition, Cisco Systems Inc., Fifth printing 2002

[28] RBridges: Base Protocol Specification, J. Touch and R. Perlman, 2006
http://ietf.cnri.reston.va.us/internet-drafts/draft-ietf-trill-rbridge-protocol-00.txt

[29] The Architecture of an RBridge Solution to TRILL, E. Gray, 2006
http://www.ietf.org/internet-drafts/draft-gray-trill-rbridge-arch-00.txt

[30] Layer 2 Bridging and Switching, K. Koymans, 2006
http://www.os3.nl/~karst/web/2005-2006/INR/Pdf/Layer2.pdf

[31] Wikipedia about Intermediate System to Intermediate System
http://en.wikipedia.org/wiki/Intermediate_system_to_intermediate_system

[32] 802.1Q standard,
http://standards.ieee.org/getieee802/download/802.1Q-2003.pdf

[33] Wikipedia about Microelectromechanical Systems,
http://en.wikipedia.org/wiki/MEMS

[34] Wikipedia about OBS,
http://en.wikipedia.org/wiki/Optical_burst_switching

[35] Considerations of point-to-multipoint route optimization using PCEMP, Jun Kyun Choi, 2006

[36] RFC 3945, Generalized Multi-Protocol Label Switching (GMPLS) Architecture, E. Mannie, 2004

[37] The practice of System and Network adminstration, T. Limoncelli, C. Hogan

[38] Digital Identity, P. Windley, 2005,
http://www.penetration-testing.com

Appendix A: License

Appendix B: sFlow sample

The sFlow Toolkit gives the following format after analyzing a pcap file:

```
startSample ----------------------

sampleType_tag 0:1
sampleType FLOWSAMPLE
sampleSequenceNo 213102
sourceId 0:1
meanSkipCount 8192
samplePool 1745731584
dropEvents 0
inputPort 1
outputPort 2
flowSampleType HEADER
headerProtocol 1
sampledPacketSize 60
headerLen 60
headerBytes ********
dstMAC 000203020082
srcMAC 0002030100c0
IPSize 46
IPTOS 0
IP6_label 0x0
IPV6_payloadLen 6
IPTTL 255
srcIP6 1b61:***:***:***:***:***:***:***
dstIP6 e173:***:***:***:***:***:***:***
IPProtocol 59
extendedType SWITCH
in_vlan 501
in_priority 0
out_vlan 501
out_priority 0

endSample   ----------------------
```

Appendix C: GMPLS Architecture

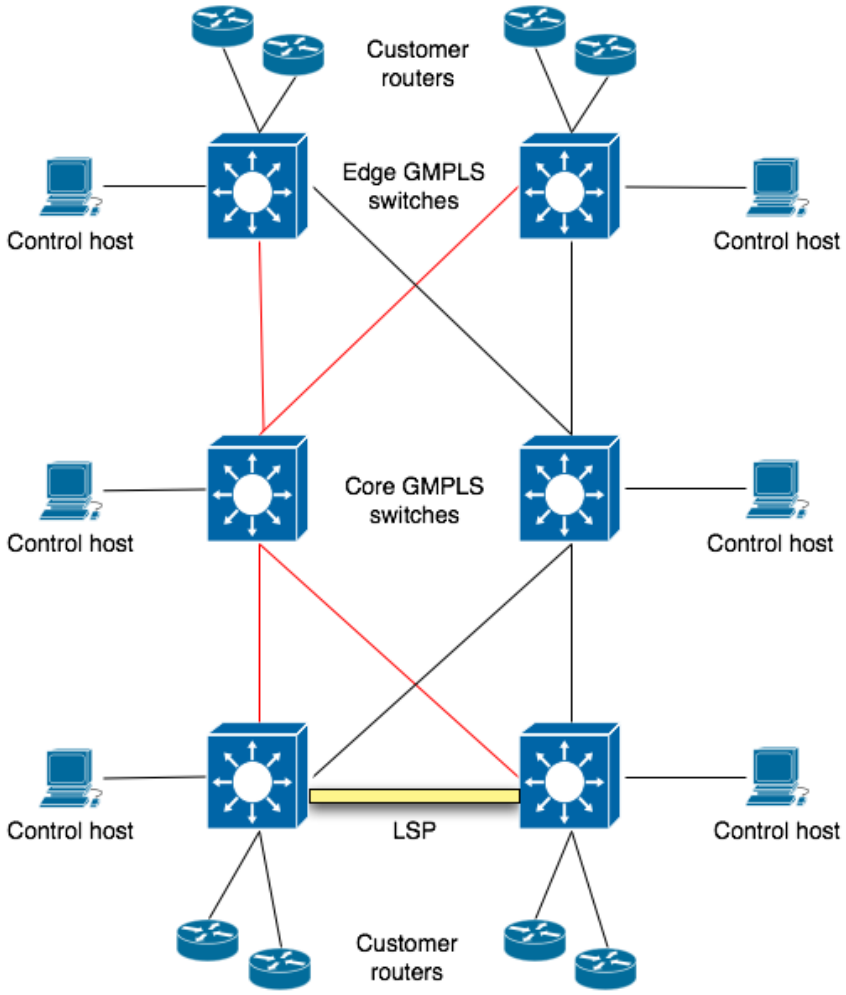


Figure 14: GMPLS Architecture

The figure shows two different networks. The black links are part of the regular layer two data network. The red links represent the GMPLS network. The yellow link represents the LSP (Label Switching Path). The control hosts communicate using OSPF to get full knowledge of the GMPLS topology. The control hosts determine the most efficient path between two switches.

All regular layer two traffic is forwarded over the black links. The control architecture determines when a flow triggers the creation of a LSP. The control architecture needs to determine the involved edge switches. The switches finally configure the LSP with the help of the control hosts.

The LSP is the logical GMPLS tunnel between two edge switches. Physically the traffic will travel from the ingress edge switch through the GMPLS core switch to the egress edge switch over the red link. This solution can roughly be compared to our proposed draft in paragraph 4.7. GMPLS also needs a control architecture to trigger the cut-through path. Both solutions determine how to forward a frame based on the destination MAC address. GMPLS uses labels to forward and distinguish cut-through traffic and our proposed draft uses MAC filters, frame filters and VLAN encapsulation to forward and distinguish cut-through traffic.