CineGrid on a PBT enabled network

Ralph Koning (ralph@science.uva.nl)

February 27, 2009

1 Introduction

CineGrid[1] is an organization which experiments with the distribution of very-high-quality digital media. Currently all high quality CineGrid content is mainly streamed over optical circuits because of their specific properties (low latency, fixed bandwidth, low jitter). CineGrid regularly uses links which are in hands of the GLIF[2] community.

There are also techniques to create layer2 Ethernet links with almost the same characteristics as optical circuits. Layer2 Ethernet is very common nowadays so being able to create these type of links adds a lot of flexibility to the distribution of CineGrid data.

2 Problem Description

Deterministic qualities are highly important for this application because every network hick-up can be seen in the video output. The application must also be tuned to get optimal performance, so low latency and constant round-trip-times are also very important for a proper stream.

At the GLIF meeting in September 2007 the organization was unable to arrange a full optical link to the location of this meeting, and they had to cross the *last mile* through the University's routed network. Because this network segment was shared with regular internet traffic people could see numerous glitches and drops in the video stream.

Also a lot of manual labour is involved to set-up a path like this. People of various organizations have to work together and communicate in order to properly configure their equipment. This is of course very error prone and it would be an improvement to (partly) automate this.

[1] `http://www.cinegrid.org`
[2] `http://www.glif.is`

3 Research Questions

Consideration of the problems mentioned in the previous section resulted in the following research questions:

Are layer2 traffic engineered paths a suitable solution for high-end media transport networks like CineGrid and do they solve some of the current problems?

- Which techniques are available nowadays to engineer layer2 paths in the network core?
- How does Provider Backbone Transport (PBT) work, what does it take to implement this for permanent and temporary use?
- Is the Quality of Service (QoS) sufficient enough for high-quality digital media streams and do the failover mechanisms converge quick enough to prevent framedrops and glitches?
- Is it possible to use Provider Link State Bridging (PLSB) for rapid distribution of this content to multiple locations and which additional requirements are needed regarding to CineGrid content?
- What is the minimum of topology information needed to set-up a suitable path across multiple domains?
- What are the consequences of using a hybrid path (e.g. combination of optical path and a layer2 path)?

4 CineGrid

The mission of CineGrid is:

> *To build an interdisciplinary community that is focused on the research, development, and demonstration of networked collaborative tools to enable the production, use and exchange of very-high-quality digital media over photonic networks.*

The University of Amsterdam focuses on building a CineGrid content exchange in Amsterdam. This involves creating an easy-to-use and accessible storage facility which is connected to multiple display and demonstration sites across the world.

The majority of the content on de CineGrid node consists of videos in High Definition (HD), Super High Definition (SHD) and 4K quality. 4K video has a resolution of 4096x2160 pixels and an uncompressed stream requires a bandwidth of approximately 7 gigabit (Gbit) per second. Currently we mainly use compressed streams which require about 1.1 Gbit.

4.1 Infrastructure

Most of the CineGrid data in Amsterdam is stored on the two Thumpers[3] which are mounted using NFS to the main streaming node, Node41. Node41 is connected to various international links through NetherLight [4] , has the ability to use the DAS3 computing cluster for transcoding and connects directly to the PBT testbed.

Currently we use two applications for streaming SHD and 4K content, NTTs multicast streaming server and SAGE[5]. The first application provides better compression in jpeg2000 and sound but also requires special hardware and a license for playback. With SAGE you can use dxt compression which is inferior to jpeg2000 but can be decoded on regular video cards, also the SAGE source code is available which allows us to make adaptions and write new software for it. Sound support in SAGE is currently being worked on.

[3] Sun Fire X4500 Servers with 48TB storage

[4] `http://www.netherlight.net`

[5] Scalable Adaptive Graphics Environment: `http://www.evl.uic.edu/cavern/sage/index.php`

4.2 Portal

The portal is developed to hide the underlying complexities of setting up network connections, content selection and controlling the video stream. The user just has to select the content and choose the right display to stream to. It's built in Python using the Django framework which allows us to easily extend it.

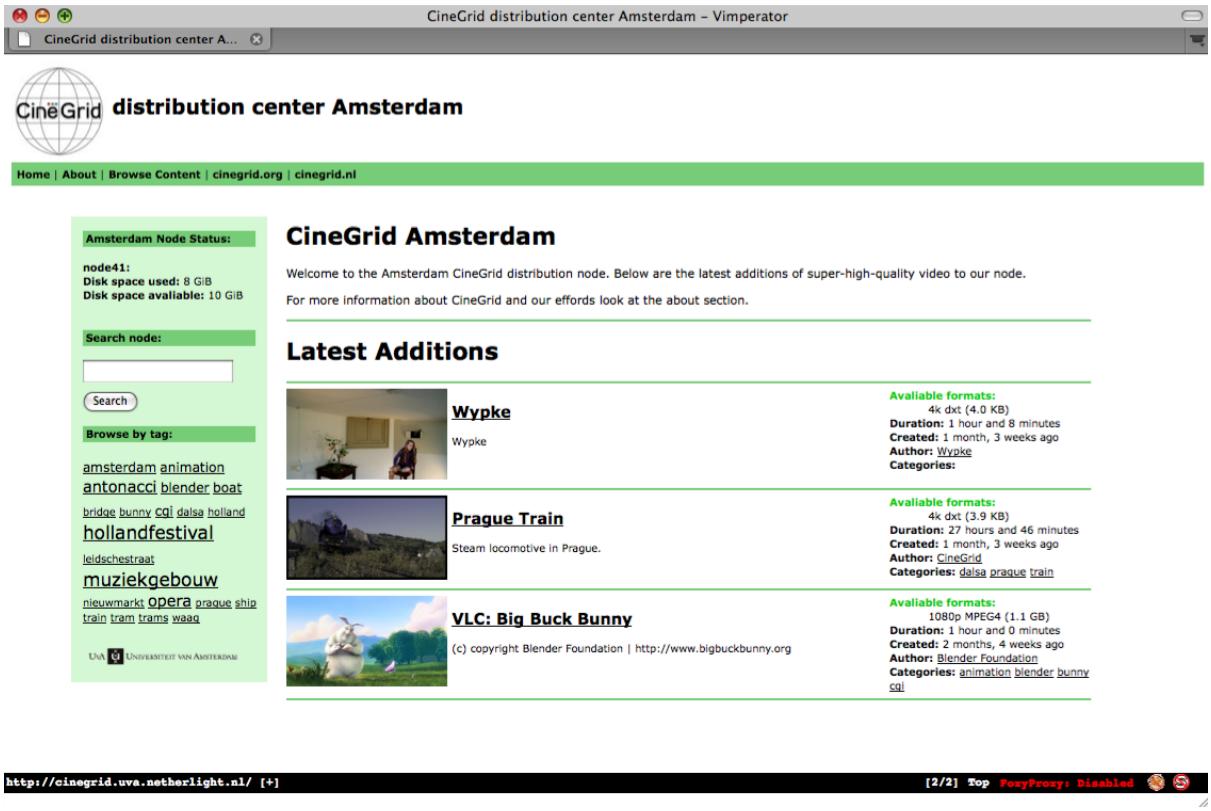

Figure 1: The CineGrid portal

Functionality is still being added to the portal, like the selection of a streaming node and automatic updates of the available content.

Behind the portal there are some other applications required to send a stream (figure: 2).

streamserver

This is the application which actually controls the stream. The user selects a video play, the portal signals the streamserver which in turn queues the request and handles the control of the stream. It is currently able to support streaming through VLC and through bvmplayer and can be extended to support more.

We are also working on a newer version of this which also is able to handle content listings and providing meta data so the portal can dynamically update its information with the information provided by the streamserver itself.

bvmplayer

One of the limitations we encountered in the SAGE environment was that the video player needed physical file access. So we asked a student to work on a prototype player that could read from a stream.

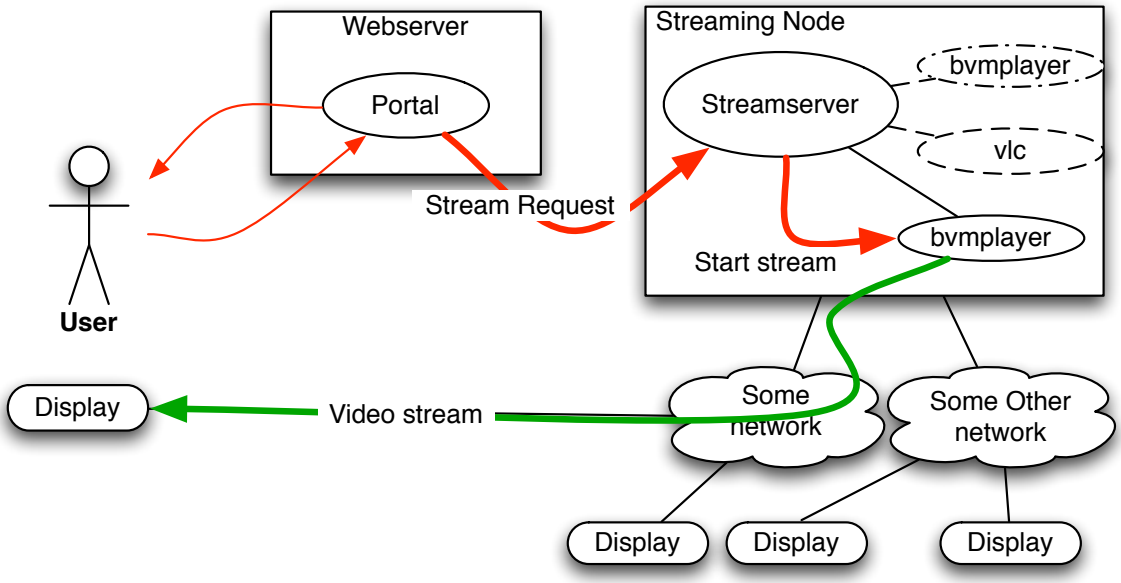

Figure 2: Portal interaction

(appendix C) A stream based player gives some more flexibility because you can send data over regular network sockets and in the future maybe a way to stream a video from two low bandwidth locations and combine this to a high bandwidth stream.

This player is now used as the default player when streaming from the CineGrid portal because it's designed to be simple and responsive. It was used for all the demo's we did this year and proved to be able to handle and stream video content up to 4k.

5 Provider Backbone Transport

PBT is a technique which enables a carrier to establish end-to-end paths over a layer 2 Ethernet network to provide the user with similar characteristics as other layer 2 techniques like SDH and SONET.

Ethernet is one of the most widely used layer2 protocols nowadays. The biggest disadvantage is that it creates one broadcast domain. To reduce this people started using vlans to separate different types of traffic but there were a limited amount of vlans, and layer2 devices needed to learn a lot of mac addresses.

Provider Backbone Bridges (PBB) solved a lot of these problems by encapsulating these Ethernet frames in another Ethernet frame and thus creating a separate network which doesn't need to know anything about the traffic it's carrying, reducing the amount of mac addresses that has to be learned. It also increased the number of identifiers for different types of services.

PBT extended on this by removing the ability to learn mac addresses in the core and only allow to create point to point links. With this kind of approach one can disable the Spanning Tree Protocol (STP) because traffic can only go in the direction specified. Also things like configuring Quality of Service (QoS) can be become much easier because one knows how the traffic flows in the network. On top of this Connectivity Fault Management (CFM) adds the ability to have protected links and statistics about the health of a link.

5.1 Testbed

The testbed at the UvA consists of the following hardware:

- 1 MERS 8600 semi-production (MERS 4.2.1.0OE)
- 2 MERS 8600 real testbed (MERS 4.2.1.0OE)
- 4 Nodes of the Rembrandt cluster for traffic generation with 10GE interfaces
- 1 Node of the Rembrandt cluster for streaming HD video.
- Node41, the main streaming node which is capable of doing 4K video.
- 1 mac mini attached to a HD display for receiving video.

Original setup

In figure 3 you can see the original version of the testbed. You can see the MERS 8600 switches called Houdini, Kazan and Geller. All switches have three 10GE ports available for PBT experiments and Kazan and Geller are also equipped with a 48 ports 1G card.

The circles with the numbers on it are ports on our optical switch and can be attached to hosts of the Rembrandt cluster on their 10GE interfaces. The connections between switches are also 10 Gbit except for the link between Kazan and Geller.

This is done on purpose because the video stream we use is about 270 MBit and its much easier to fill a 1 Gbit connection than a 10 Gbit connection.

The green line shows the video stream flowing from rembrandt4 to the mac-mini and the display and the red line shows the flow of random traffic which is going to compete with the video stream.

QoS tests on this testbed turned out to be negative. It seemed that with correct configuration the switches did not enforce QoS at all on 1G link. It turned out the 8648GTR card was not PBT compatible (figure 4).

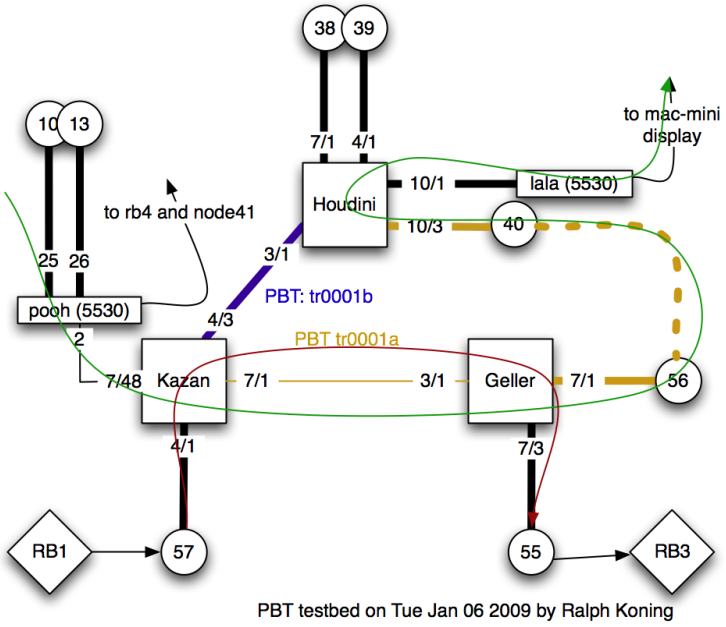

Figure 3: Original version of the PBT testbed

3.3.7 MERS 8600 Line Cards

The latest MERS features are focused on the top 9 line cards below: ESM, R and RC-modules.

Card	Ports	Ethernet VPN UNI support	Ethernet VPN NNI support	Customer IP VLAN support (Note 1)	L3 VLAN support (Note 2)	L2 VLAN support (Note 5)	PBT Trunk	PLSB	ESU Ring
8668 ESM	8-port GE SFP	Yes[9]	No	Yes	Yes[2]	No	No	No	Yes
8648GTR [8]	48-port 10/100/1000 RJ45	Yes	No [6]	Yes[4]	Yes	Yes	No[7]	No	No
8630GBR	30-port GE SFP	Yes	Yes	Yes[4]	Yes	Yes	Yes	Yes	Yes
8683XLR	3-port 10GE XFP LAN	Yes	Yes	Yes[4]	Yes	Yes	Yes	Yes	No
8683XZR	3-port 10GE XFP LAN/ WAN	Yes	Yes	Yes[4]	Yes	Yes	Yes	Yes	No
8648GBRC [11]	48-port FE/GE SFP	Yes[10]	Yes[12]	Yes	Yes	Yes	No	No	No
8630GBRC [11]	30-port FE/GE SFP	Yes[10]	Yes	Yes	Yes	Yes	Yes	Yes	Yes
8606XLRC [11]	3-port 10GE XFP [13]	Yes[10]	Yes	Yes	Yes	Yes	Yes	Yes	No
8626XGRC [11]	24-port FE/GE SFP + 2-port 10GE XFP [14]	Yes[10]	Yes[12]	Yes	Yes	Yes	Yes[15]	Yes[16]	No

Figure 4: Hardware compatibility chart

Final setup

Due to the compatibility problems with the original setup the testbed needed to change. This involved making all connections 10 Gbit because these cards were capable of doing PBT correctly. An disadvantage of this is that the nodes in the Rembrandt cluster were only capable of sending about 6 Gbit of traffic on their 10 Gbit interfaces. This means 4 nodes are needed to fill a link.

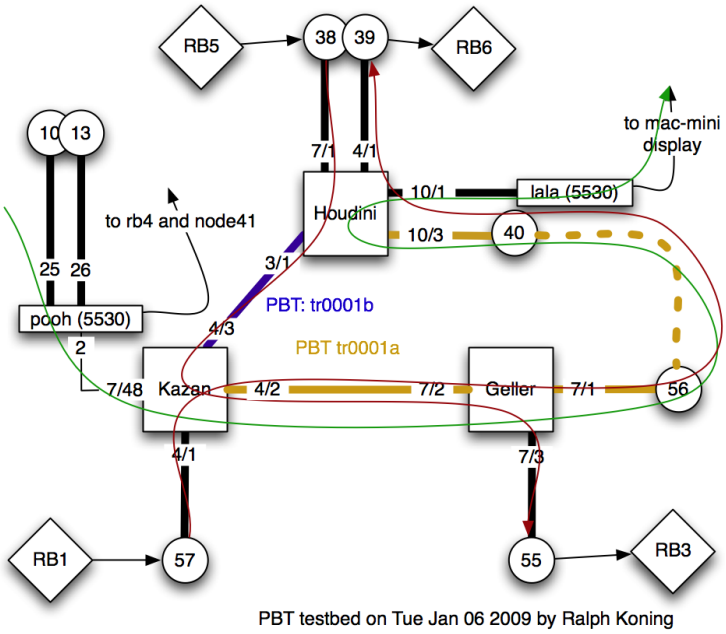

Figure 5: Final version of the PBT testbed

*

So we made the link between Kazan and Geller 10 Gbit and connected two more rembrands to Houdini (figure 5) The red lines show the random traffic again with the video stream in green. In this situation the link between Kazan and Geller should be congested if every node is sending simultaneously.

Because the random traffic is not sent over an PBT trunk and this topology has some loops in it some spanning-tree groups had to be made to make STP behave correctly, which of course adds to the complexity of this setup.

5.2 QoS

These tests are done on the 10 Gbit link between Kazan and Geller (figure 5), if every node is sending this link gets congested. A graph of the video stream shows a huge drop in bandwidth if this happens as you can see in figure 6. The video used for this experiment is a CineGrid clip called De Waag.

QoS was tested by allocating three percent of the 10 Gbit bandwith (300 Mbit) for the video stream and putting this traffic in a high priority queue. The other traffic is able to use up to 100 percent if the stream isn't running but as soon as it starts the it will be limited to 97 percent. So the video stream is left unharmed.

With QoS enabled and using the same traffic pattern as in figure 6 the video stream remains stable as you can see in 7

As you can see the stream looks rock solid except for the small drop around the 45 seconds. Which is a small hickup probably caused by one of the hosts and also noticeable in graphs of streams when there

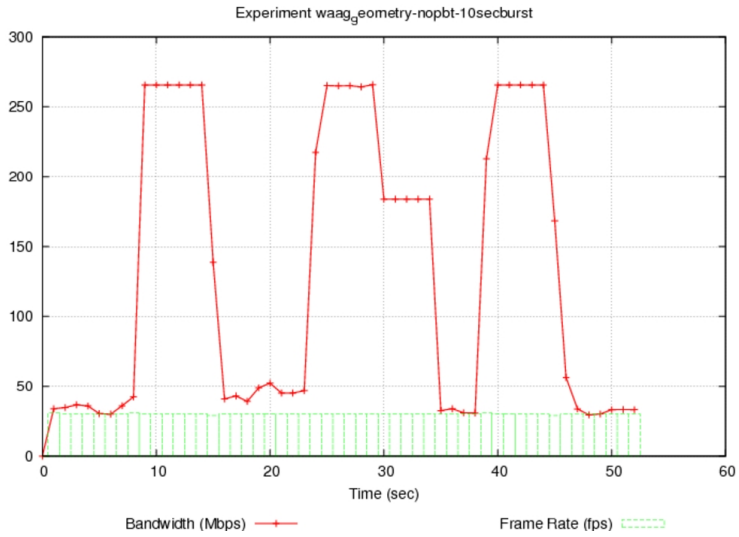

Figure 6: Stream performance without QoS with 10 second intervals of traffic

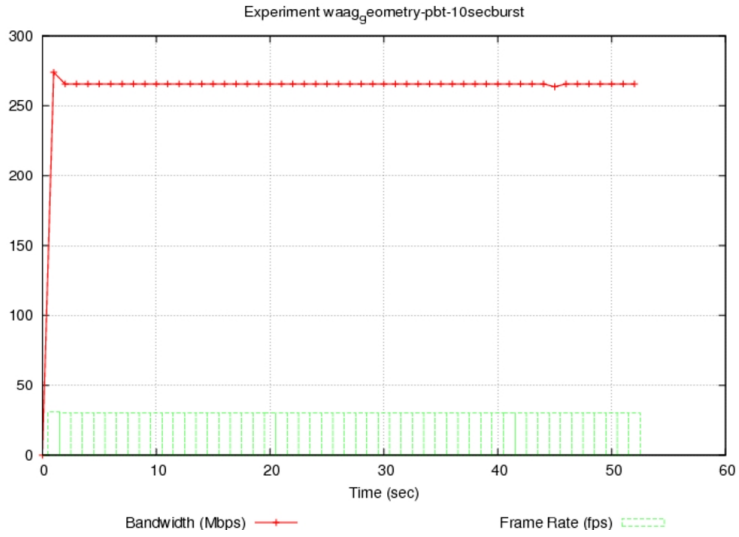

Figure 7: Stream performance with QoS with 10 second intervals of traffic

is no other traffic. It isn't noticeable in the video stream itself so it could also an inaccuracy in the bandwidth reports from the SAGE application.

5.3 Link failover

Another aspect of PBT I wanted to look at was link failover. As we can see in figure 5 we have two trunks, the orange one `tr0001a` and the blue one `tr0001b` which is the secondary trunk. The ideal situation is that if the primary trunk `tr0001a` fails `tr0001b` can take over the stream flawlessly.

The health of a PBT link is monitored using CFM (Connectivity Fault Management) messages. These messages are send over the link at a configurable interval with a default of 10s. If three of those messages are lost within a timeframe of about 30 seconds the link failover will be triggered.

This results in 30 seconds of link outage and a disturbance in the video stream running on top of it (figure: 8). This not acceptable in most of the CineGrid demo's.

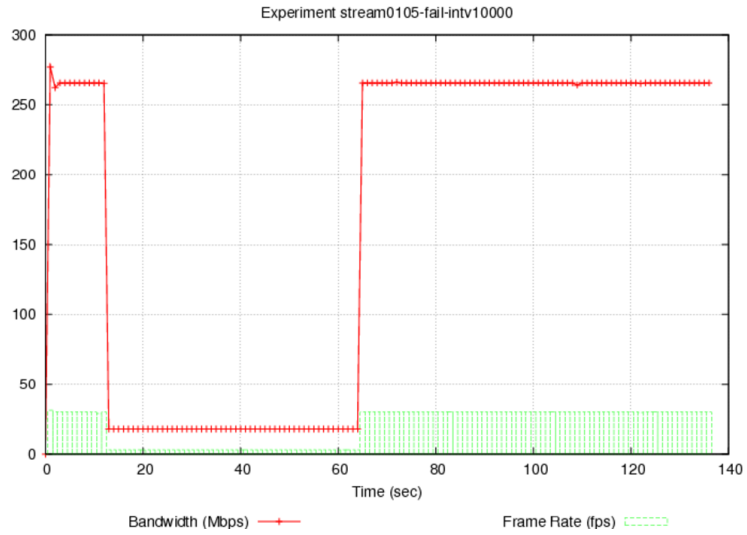

Figure 8: Failover with a CFM interval of 10 sec

Lowering the check interval does improve the flow, at 100 msec this results in a small but visible hickup and at its lowest 10 msec it results in a small drop in the graph but no visible results in the video stream as you can see in figure 9.

Note that these tests are done within a local testbed, if you use international links round trip time can increase which means that there is a bigger delay in the sending of CFM messages which could result in slower response to link failure which could even at the lowest setting be noticeable in video streams.

5.4 Other experiences using PBT

Working with the technique and administering this small testbed already provided the need for a management tool because configuration needs to be done at a lot of different places in the network. This is a tedious job to do and still requires a lot of information on the network and PBT.

Also lack of support from other vendors is a huge setback in a multi-vendor network environment which means PBT is only usable in networks which already have a certain amount of Nortel devices. From an administrative point of view this makes PBT only suitable for a long term solution.

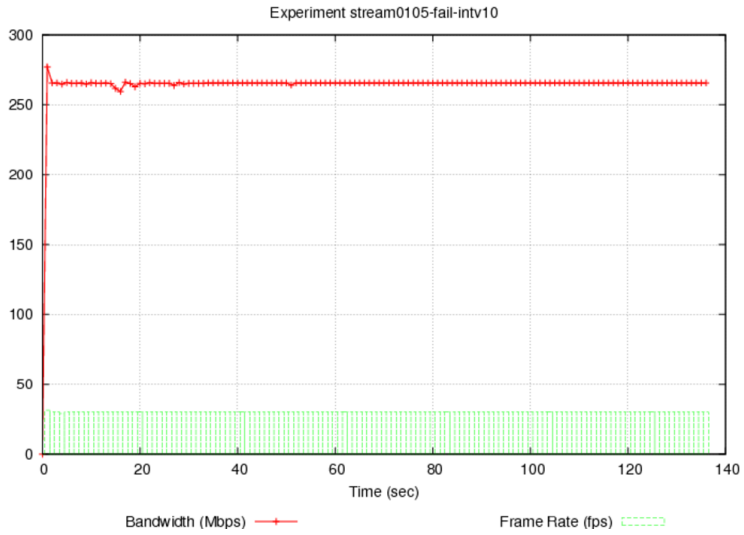

Figure 9: Failover with a CFM interval of 10Msec

6 Demos

There were also some demos regarding PBT and CineGrid which contributed to the research.

6.1 ATS

The demos for the ATS involved streaming 4K video from the servers at the UvA to the tiled display at Nortel over a 1 Gbit lightpath from Amsterdam to Ottawa and a presentation. At the same time a lot of traffic was injected in the PBT testbed to congest the link and demonstrate that the QoS would hold.

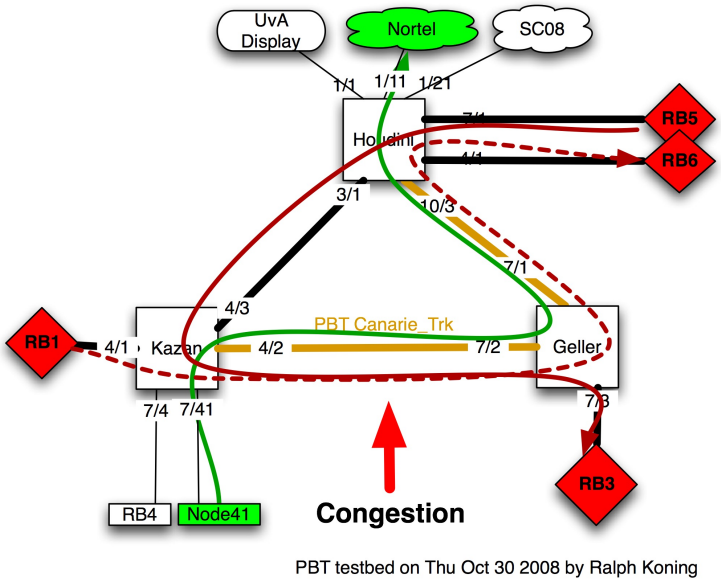

Figure 10: ATS Demo setup

For this demo we also extended the PBT testbed at the UvA over the lightpath to a MERS in Ottawa.

Because it was only one path we could not demonstrate things like failover and we had to change part of the setup at the UvA side (figure: 11)

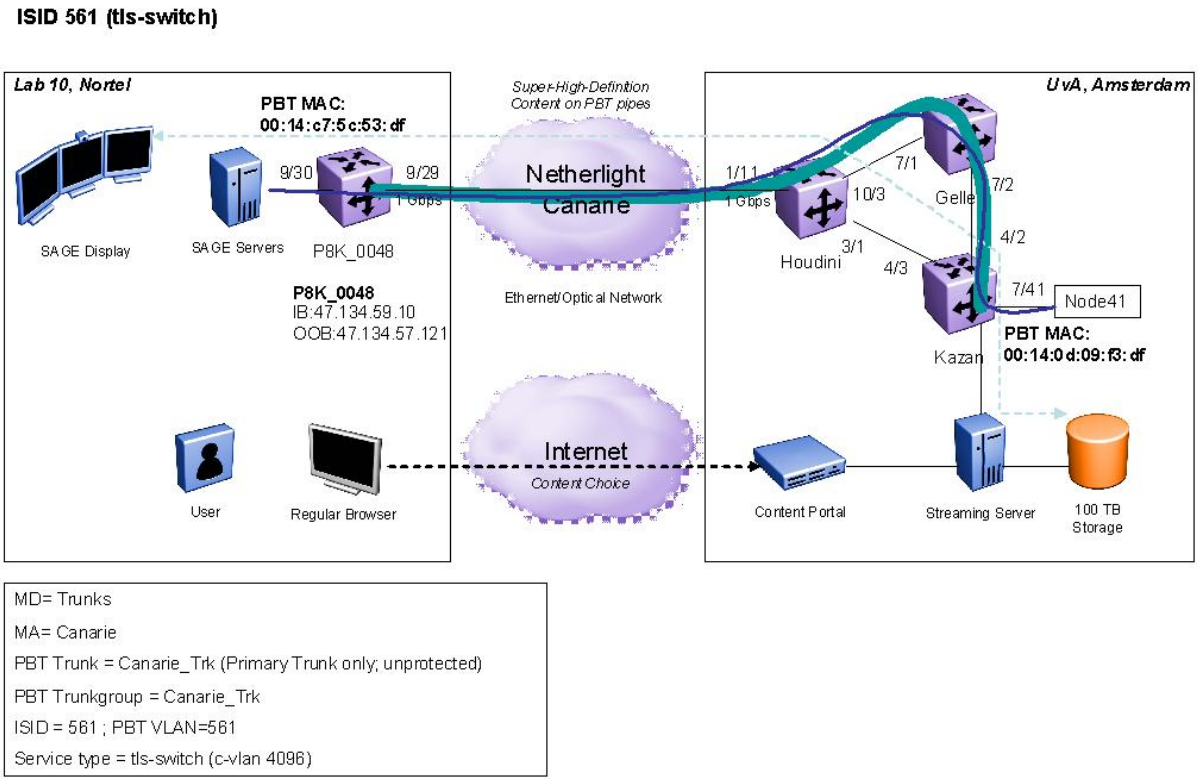

Figure 11: ATS Demo setup

We had to use Node41 for streaming in stead of Rembrandt4 because Node41 is attached to the CineGrid storage network and Rembrandt4 doesn't have the capability to send 4K video. We also had to create an new I-SID which goes al the way up to the Nortel MERS called `PBT_Canarie_Trk`

The result was a stable video stream to the tiled display without any framedrops. Due to the 1 Gbit link we were only limited to stream about 27 frames per second which is less than the 30 fps required for some of the CineGrid content so the video was a bit slower.

6.2 SC08

For SC08 we reverted back to the final setup (figure 5) to have two trunks and being able to demonstrate link failover. We also used HD content again which of course lowered the streaming bandwidth.

So we looped a HD stream coutinuously from the UvA to the HD display at SC08. Again we injected bogus traffic into the network. On a PC next to the HD display was an application running to alter the testbed. People had the ability to toggle QoS on the PBT link and the ability to shutdown a port in the testbed to trigger a link failover. The user could also see realtime bandwidth graphs of both the bogus traffic flows and the video steam so they could see the congestion on the link when QoS is turned off.

We also used the CineGrid portal to control three displays. A user could select content and stream it to the tiled display of SARA[6], to the CANARIE[7] booth for HPDMnet[8] and through the PBT testbed to the HD screen.

[6] `http://www.sara.nl`
[7] `http://www.canarie.ca`
[8] `http://www.hpdmnet.net`

7 Conclusions

Because of the widespread use of Ethernet PBT is a valuable extension for circuit like behavior. Although the configuration might be a bit hard it can certainly be improved by a good management tool. I won't recommend PBT for temporary solutions but it's certainly useful as a permanent solution in large networks.

For use within CineGrid PBT is an interesting technique. Because using QoS on these links allows for stable and reliable video streams. But because the amount of partners, the heterogeneous and dynamic environment and lack of support of other vendors I think it's only useful for some of its members.

8 Future Work

Because a lot of organizations participate in CineGrid it's very useful to look at the multi domain aspects of PBT, how this works if the other party is running a different network technoligy in his core and how to easilly identify these multi-domain links.

PLSB is another interesting technique giving the ability to use point to multipoint connections and maybe the ability to multicast video across these networks. (appendix C)

The ability to use Network Description Language[9] (NDL) to model a PBT network and some tools to easily use this information for the management of the network would also be interesting and could ease the deployment of this technique. This also allows for better interaction between customer applications and their networks, and allows a application to control the network e.g. the ability to setup links when required using a portal.

[9] `http://www.science.uva.nl/research/sne/ndl`

A Timeline

11-27	Project plan.
01-25	First version of portal online.
02-12	Student starts bachelor thesis: *CineGrid: storing and streaming 4k video content"*
02-14	SURFnet/Nortel partner visit.
02-15	Presentation on Portal and informal discussion about the project.
02-21	Document with PBT scenarios.
05-10	Conference call.
05-06	Received requested hardware for testbed.
05-13	First formal 4k demonstration operated by UvA/SURFnet at SURFnet relatiedagen.
05-16	Hans started to work on building a bandwidth/latency test suite to test QoS aspects.
05-21	Presentation on CineGrid and PBT at Terena Networking Conference 2008.
05-28	Presentation on CineGrid progress at Research on Networks meeting.
06-02	Start master student project: *Streaming and storing CineGrid data: A study on optimization methods.*
06-02	Start master student project: *Multicast in a CineGrid testbed.*
06-05	Testbed up and running.
07-02	Presentations on student projects.
07-15	Conference call.
08-01	Talk with Jan Willem Elion from Nortel Netherlands about possible ATS demo.
08-06	Got support from Martin Williams for the ATS demo.
08-29	PBT Mid-Year report
08-01	Lightpath request to Ottawa
09-17	Got support from Matthew
09-18	Call with Matthew and Inder
09-23	Figured out that QoS problems exist due to PBT incompatible hardware
09-23	Lightpath from UvA to Ottawa realized
09-24	Requested 8630GBR cards from various parties instead of the incompatible 8648GTR
10-09	Successful 4k streaming to Ottawa trough PBT Testbed
10-10	Successful 4k streaming from CineGrid portal
10-15	Recieved one 8630GBR card, used to terminate link to Ottawa
10-20	Portal update to support multiple display nodes and VLC streaming
10-30	PBT Testbed extended with a trunk to Ottawa
11-04	ATS Demos
11-15	ATS Demos finished
11-17	Reconfigured testbed for SC08 PBT Demo and HPDMnet demo
11-21	SC08 Demos finished
12-08	Upgraded portal to Django 1.0
12-11	Removed link to Ottawa

B Presentations

Nortel SURFnet parner visit (02/14/2008)

An informal presentation on the first version of the CineGrid portal and how we plan to use this for future demonstrations.

SURFnet relatiedagen (05/13/2008)

First formal 4k presentation and demo completely operated by people from the UvA and SURFnet with the result that we know how to operate the Keio DMC setup using the NTT codec without any help.

TERENA Networking Conference 2008 (05/21/2008)

A presentation titled: "CineGrid over PBT paths:High quality digital content over L2 engineered path" in the "Making grid applications happen" track covering both CineGrid and PBT and how CineGrid can benefit from PBT.

Slides: `http://tnc2008.terena.org/core/getfile.php?file_id=429`
Video Stream: `http://tnc2008.terena.org/schedule/presentations/show.php?pres_id=22`

Research on Networks (05/28/2008)

This presentation was a status update on CineGrid and related projects. It also covers PBT, the portal, and a way of playing 4k using file streams.

C Reports and technical documents

This are reports of related research done at the University of Amsterdam, my role was to supervise this projects.

Bachelor Thesis: ***CineGrid: storing and streaming 4k video content"***

Sander Knopper worked on a player for 4k video content which can read from a stream and send it to a SAGE display in stead of requiring physical file access. There are also some benchmarks in it.

Master Thesis: ***Streaming and storing CineGrid data: A study on optimization methods.***

Sevickson Kwidama looked at the behaviour, bandwidth and CPU load of HD video streamed to SAGE. He also did some performance tests for GlusterFS a cluster file system we may use sometime to store video content.

`http://staff.science.uva.nl/~delaat/sne-2007-2008/p04/report.pdf`

Master Thesis: ***Multicast in a CineGrid testbed.***

Igor Idziejczak looked at different traditional multicasting methods PBT and PLSB for a way to multicast video traffic. He also looked at SAGEbridge, a part of the SAGE environment which may provide another way to do multicast video streams at the application layer.

`http://staff.science.uva.nl/~delaat/sne-2007-2008/p25/report.pdf`