

Project IPTV Platform

VAP RSM 2010-2011



Anass Touati
Mehdi Masbahi
Claudia Elena Toader
Jose Pablo Loera Saucedo

Tutors: Hossam Afifi
Emad Mohamed ABOUSABEA



Table of Contents

I. General concepts IPTV	3
IPTV Project.....	3
II. DVB (Digital Video Broadcast)	5
1. MPEG.....	6
2. DVB-S.....	6
3. DVB-C	7
4. DVB-T	8
III. VLC (VideoLAN Client)	9
Unicast streaming.....	9
VOD streaming	10
IV. Set-Up-Box (STB).....	11
1. The Linux4.TV Set-Top Box Open Source Project.....	11
a. Graphical Windowing System.....	12
b. Kernel and Device-Driver Layer	13
c. Video Middleware Layer.....	13
Audio Subsystem.....	13
Video Capture Subsystem	14
Overlay Subsystem.....	14
MPEG Subsystem	14
TV Subsystem	14
d. WebMedia and Applications Layer.....	14
2. SET-TOP-BOX OPEN SOURCE HARDWARE/SOFTWARE	16
a. Set top box open source de Neuros.....	16
b. VT900 Set-Top Box (STB): Century Embedded Technologies and VT Media Technologies.....	16
c. Boxee Set-Top Box and Remote	17
d. Review: Logitech's Google TV box	17
e. The YBox2	18
f. GeexBox	18
V. MythTV.....	19
1. Introduction.....	19
2. MythTV architecture	21
3. Hardware.....	22
4. Detailed Comparaison Chart	23
5. Installation.....	24
a. Before installation	24

b. From source	25
c. From package (MyUbuntu)	26
d. Configuration	26
VI. Experiments.....	28
VII. IPTV Multicast.....	33
1. Introduction	33
2. Multicast routing protocols	34
i. PIM-SM (Protocol Independent Multicast Sparse Mode)	35
ii. IGMP	36
i. IGMP v1	36
a. IGMP v2	36
b. IGMP v3 and IGMP snooping	37
VIII. Conclusion	39
Bibliography	40
Schedulers	41

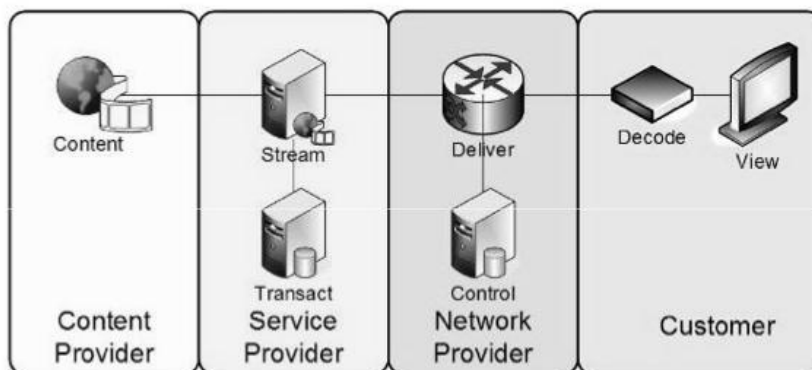
I. General concepts IPTV

Internet Video is video content over the Internet open public network. Over the web, portals offer video streaming and discrete content elements, no Quality of Service and a wide range of video formats & streaming protocols.

On the other hand, IPTV has become a high-benefit service for the operators by adding multimedia content delivery (TV, Video & Audio) over their own network. One hop before the client they include a Set Top Box to give interactive and customized services that add aggregated value to the service like:

- Targeted advertisement
- Live stats of program watching
- Network PVR (offline TV programs)

Content flow for Internet video

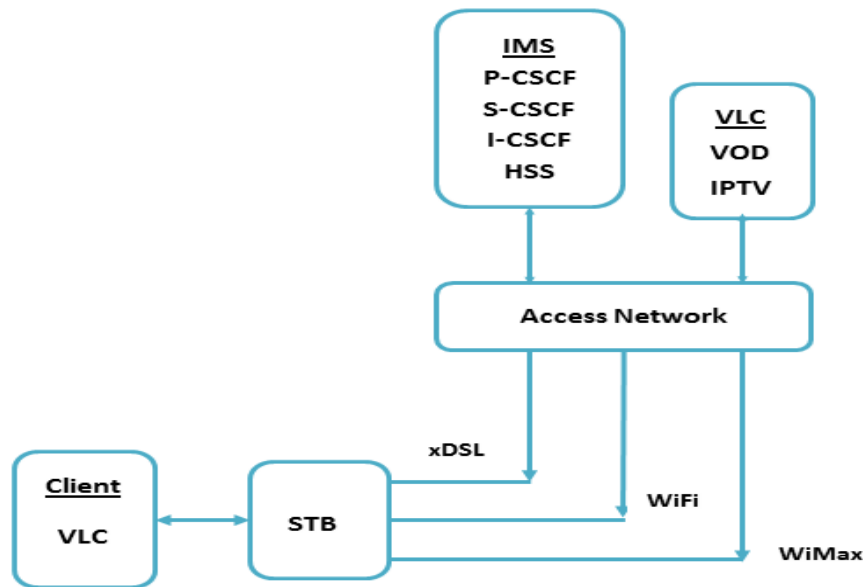


* Ref: ITU TSB IPTV Consultation meeting (Doc. Iptv018e and 20e)

The process of live streaming involves a camera for the video, an encoder to digitalize the content, a video publisher where the streams are made available to potential end-users and a content delivery network to distribute and deliver the content. The media can then be viewed by end-users live.

IPTV Project

Our project is based on a proposed typical architecture for video streaming over a wireless network and a classic ADSL network. Their functionalities will be tested in order to verify limitations, quality and available bandwidth.



Proposed IPTV Architecture

Project's Main steps

The project was developed around 3 main points:

- **Research on Digital Video Broadcasting:** in our case DVB-T standard is used because it's the most widely deployed and well adapted to digital streaming. The research was mostly done to know what we were dealing with and also the techniques in which data is transmitted. This part was roughly an introduction and an approach to the subject so that afterwards we could continue with the implementation.

- **Installing VLC software** under Linux (Ubuntu) in two PC's and testing video streaming between them as to simulate the path between server and client. We achieved the data streaming by using UDP, TCP, RTP and even HTTP protocols. Depending on the application we want each protocol is useful for certain things for example, Video on Demand is a unicast transmission established by the server and because RTP is designed for end-to-end, real-time and transfer of data streams it is chosen. For live, multicast video streaming in which a huge number of subscribers are joined then UDP is compatible for giving the service with a simple transmission.

- **Research about source Set Top Boxes:** Here we found several virtual boxes but which were used in certain projects for their own specifications and also physical ones from ComTrend. These last ones were not useful because ComTrend customizes the boxes specifically for its customers (big enterprises) and they are hard to configure by someone external to the company. So after continuing to look for virtual boxes we found and installed MythTV and GeexBox to start testing.

II. DVB (Digital Video Broadcast)

DVB is a European open standard for digital television. It led by the DVB project, an international industry consortium. There are many specifications of The Digital Video cover digital services delivered over, satellite, cable, terrestrial transmitters, the internet and mobile communication systems.

DVB project is described in two parts:

- Source coding – data compression
- Transmission

The compression algorithm for the video is MPEG2 and MUSICAM for the audio.

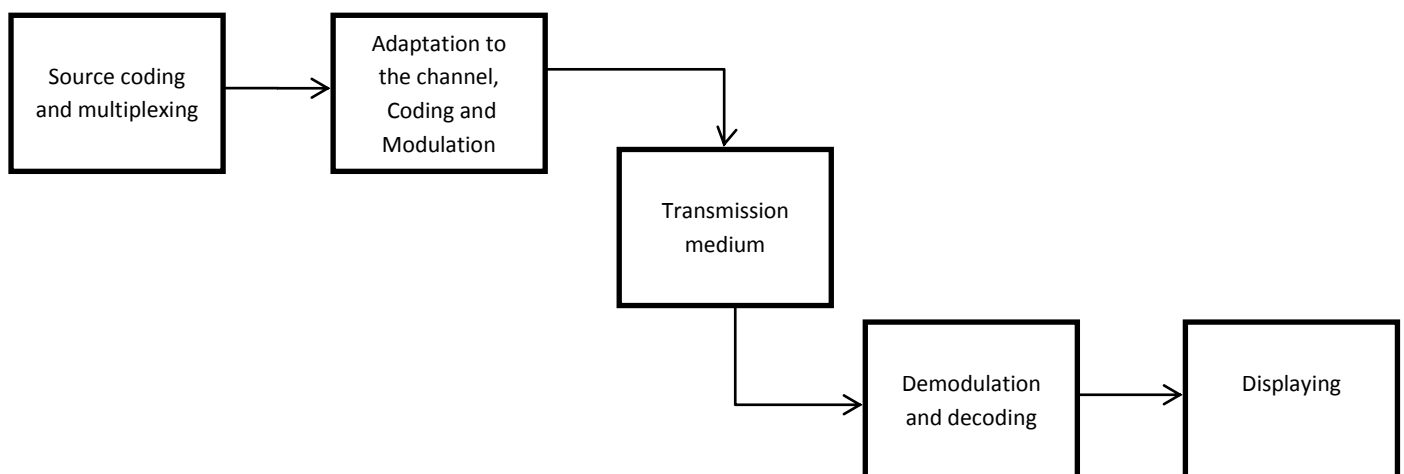
Most important transmission supports are:

- Satellite DVB-S
- Cable DVB-C
- Terrestrial DVB-T

For each one, the DVB project has chosen a specific modulation:

- DVB-S → QPSK
- DVB-C → 64-QAM
- DVB-T → COFDM

Figure 1: General architecture of DVB:



1. MPEG

The standardized technique used to send audio and video over the network in a fast and efficient way is MPEG-2. This protocol is directed at broadcast formats at higher data rates; it provides extra algorithmic 'tools' for efficiently coding interlaced video, supports a wide range of bit rates and provides for multichannel surround sound coding. MPEG-2 decoders will also decode MPEG-1 bit streams.

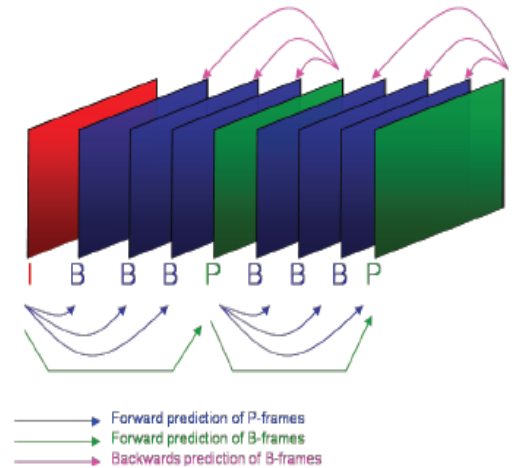


Figure 2 : prediction in MPEG

Main features of different versions of the standard:

MPEG -1

Intended for use in creating video compact discs (CDs), operating at 1Mbps. Occasionally used for: low-cost surveillance and some web video, doesn't support Interlacing or HD video and lacks of advanced coding techniques

MPEG-2

Approved in 1996, it is used widely in all video fields. Chosen by DTV broadcast TV in the USA Supports Interlacing & HD video (720p, 1080i & p). Multiplexing of a number of Video & Audio streams, offering multichannel, multilingual, and multi-program capabilities 5 channel audio (surround sound) support Advanced Audio Coding (AAC)

2. DVB-S

DVB-s describes the DVB frame structure, the modulation of the MPEG-2 packets and the channel coding system for satellite (frequency band from 11 to 12 GHz) digital Television services. It also gives minimum performance required by the system in terms of resistance to errors and noise. The implementation of the receiver part is left to the manufacturer choice. The European Standard EN 300 421 defined the main functions of the DVB-S.

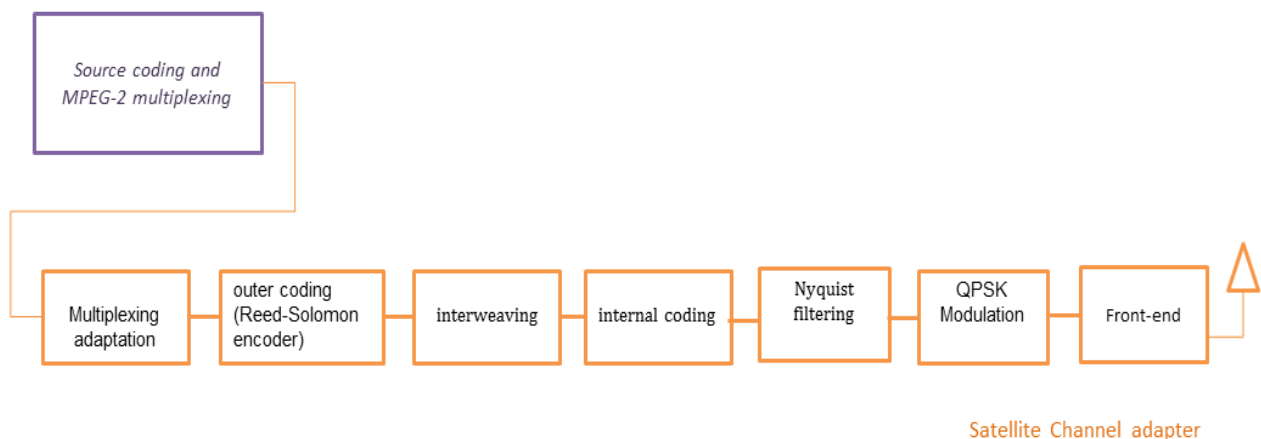


Figure 3: Scheme of a DVB-S transmission system

DVB-S2 is the second-generation specification for satellite broad-band applications, developed by the DVB Project in 2003 and ratified by ETSI (EN 302307) in 2003. It designed as a successor for the popular DVB-S. DVB-S2 is envisaged for broadcast services including standard and HDTV, interactive services including Internet access, and (professional) data content distribution. The development of DVB-S2 coincided with the introduction of HDTV and H.264 video codecs.

3. DVB-C

DVB-C stands for Digital Video Broadcasting - Cable and it is the DVB European consortium standard for the broadcast transmission of digital television over cable. This system transmits an MPEG-2 or MPEG-4 family digital audio/video stream, using a QAM modulation with channel coding.

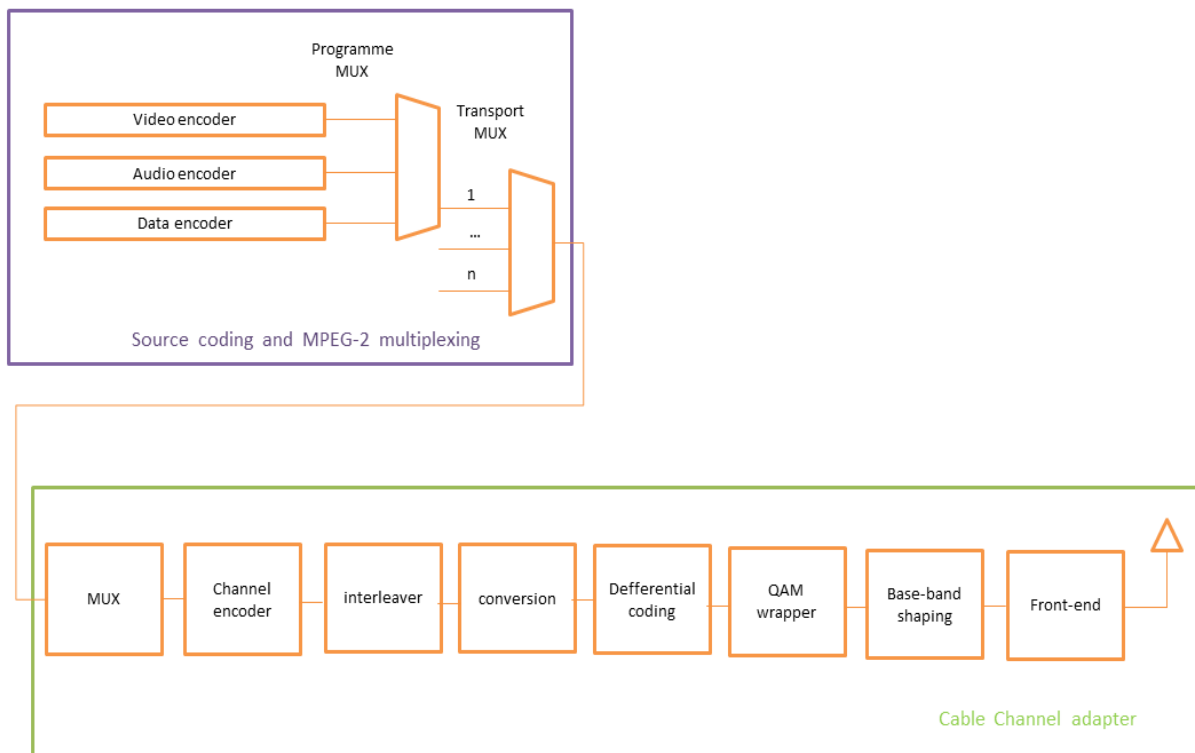


Figure 4: Scheme of a DVB-C transmission system

On February 18, 2008 it was announced that a new standard – **DVB-C2**.

Modes and features of DVB-C2 in comparison to DVB-C:

	DVB-C	DVB-C2
Input Interface	Single Transport Stream (TS)	Multiple Transport Stream and Generic Stream Encapsulation (GSE)
Modes	Constant Coding & Modulation	Variable Coding & Modulation and Adaptive Coding & Modulation
FEC	Reed Solomon (RS)	LDPC + BCH
Interleaving	Bit-Interleaving	Bit- Time- and Frequency-Interleaving
Modulation	Single Carrier QAM	COFDM
Pilots	Not Applicable	Scattered and Continual Pilots
Guard Interval	Not Applicable	1/64 or 1/128
Modulation Schemes	16- to 256-QAM	16- to 4096-QAM

4. DVB-T

It is the standard was chosen by the DVB project for the terrestrial channel, this kind of the transmission is certainly the most difficult to deployed. There is high attenuation at the receiver, signal to noise ratio (SNR) is limited to approximately 18.6 dB, and Inter-Symbol Interference (ISI) due to significant multipath long distance (high propagation). For the transmission over the terrestrial channel of 8MHz, the upper limit of binary flow is around 24 Mbit/s. The binary good put depends of the correction code and the modulation. Like the other standard of DVB, DVB-T uses the MPEG-2 for the coding of baseband signal. But it uses a new modulation OFDM (Orthogonal Frequency Division Multiplexing).

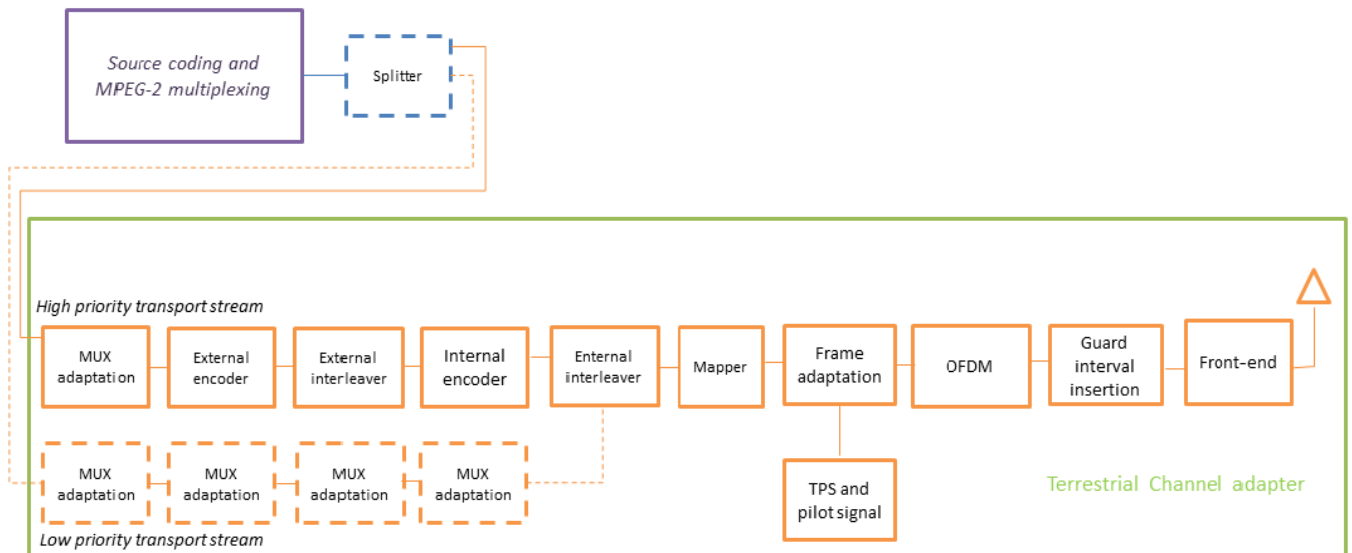


Figure 5: Scheme of a DVB-T transmission system

A new version of the standard was developed recently supporting HDTV format.

III. VLC (VideoLAN Client)

VideoLan is an open source project, leaded by a team of volunteers and developing multimedia solutions. The project started in 1996 at Ecole Centrale Paris and became an international project with developers from more than 20 countries. The software the most famous from VideoLan collections is VLC. It is cross-platform and with a huge library of codecs it can encored many audios and videos. Some features are only accessible from the command line interface.

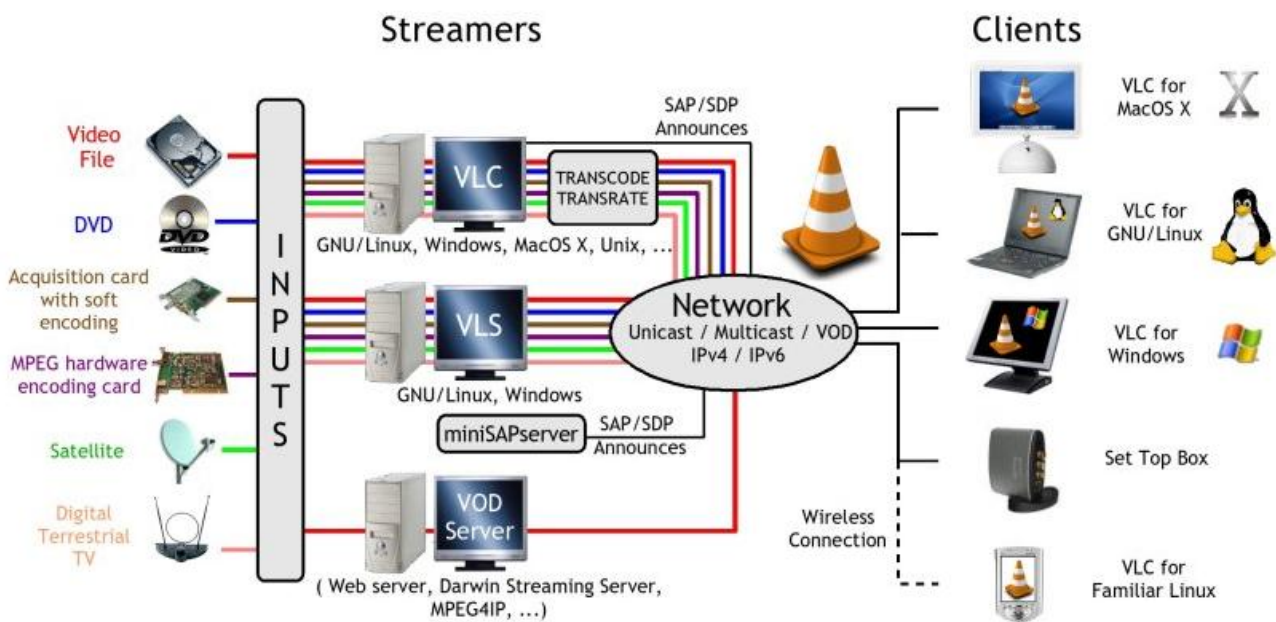


Figure 1: VideoLAN Streaming solution

VLC can collect many TV sources like satellite or terrestrial and stream them through LAN or the internet over IPv4 or IPv6. VLC is able to announce its streams using the SAP/SDP standard or using Zeroconf (also known as Bonjour). SDP is a session directory protocol and SAP is a session announcement protocol; the protocols display multicast session names and correlate the names with multicast traffic.

In this project we had to use VLC like a server streamer and a client. Essentially with Unicast (UDP, RTP) and VOD protocols (RTSP).

Unicast streaming

For this part, we used the following command:

```
vlc -vvv test.mp4 --sout udp:10.10.1.2 --ttl 10
```

Where:

- test.mp4 is the file which we want to stream

- 10.10.1.2 is the client's address
- 10 is the TTL (Time To Live) value of packets.

To receive unicast stream use the following command:

```
vlc -vvv udp:[@:port_UDP]
```

We can change UDP by other unicast protocol (RTP, HTTP).

VOD streaming

We can also use VLC to stream videos on demand. To do this, we use the RTSP (Real Time Streaming Protocol) protocol which offers essential features to control the video (pause, play ...).

The following commands show the procedure for RTSP streaming:

```
vlc -vvv -color -I telnet-password admin -rtsp-host 0.0.0.0:5554
```

Where:

- telnet starts the VLC's Telnet interface
- admin is the password to connect to Telnet interface
- 0.0.0.0 allow streaming to any host
- 5554 is the streaming port

And in a new console:

```
telnet 10.10.1.1 4212
```

Where 10.10.1.1 is the server's address and 4212 is the outgoing port.

After entering in the telnet interface, type:

```
new test vod enable
```

```
setup test input extrait.mp4
```

In the client, type the following to play the video:

```
vlc rtsp://10.10.1.1:5554/test
```

IV. Set-Up-Box (STB)

1. The Linux4.TV Set-Top Box Open Source Project

Linux4.TV is a open-source set-top box platform with support for video on demand, DVB and video streaming.

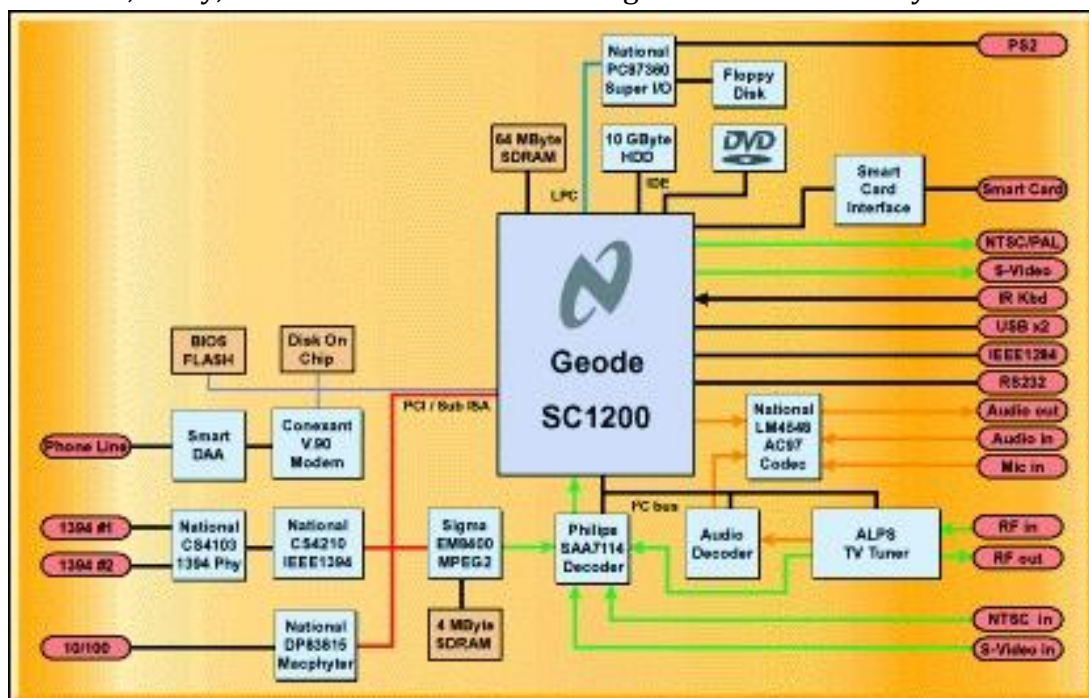
System Architecture

Linux4.TV Project uses the National's Geode SP1SC10 demonstration platform as the hardware for the first implementation.

Geode SP1SC10 technical specifications:

- CPU SC1200 – 266MHz
 - x86-compatible 32-bit instruction set
 - MMX support
 - 2-D graphics acceleration
 - integrated NTSC/PAL controller
 - CCIR-656-compatible video input port for full-screen video display
- Philips SAA7114 chip for input decoding
- Sigma Designs EM8400 chip for real-time hardware decoding of MPEG-2 digital video streams

This system design allows video inputs to be decoded and routed into the video input of the CPU so that to combine in real time browsers and other application's graphics data output with the desired video. This architecture allows also applications to be integrated with few, if any, modifications so that working with the video subsystem.



- Geode STB Platform Architecture -

The Linux4.TV purpose is to make STB technology available at a large scale by developing open source software platforms and lower-cost hardware platform in partnership with manufacturers.

There for National is developing an open source low-cost STB to replace SP1SC10. In addition, several other vendors have platforms based on the Geode SC1200, and porting of the Linux4.TV code is in process. These vendors include VT Media Technologies and Cocom Group. Advantages of third-party vendors include a variety of form factors for set-top box deployment.

Software Architecture

There are four major layers in the software architecture:

1. Kernel and device driver,
2. Video Middleware,
3. Windowing system
4. WebMedia/applications Layer

National Semiconductor contributed the kernel drivers and Video Middleware layers, with Century contributing the WebMedia user interface, windowing system, applications layer and overall distribution. The complete distribution is available both as a bootable system image and as a complete source tree. Following are more technical descriptions of each layer.

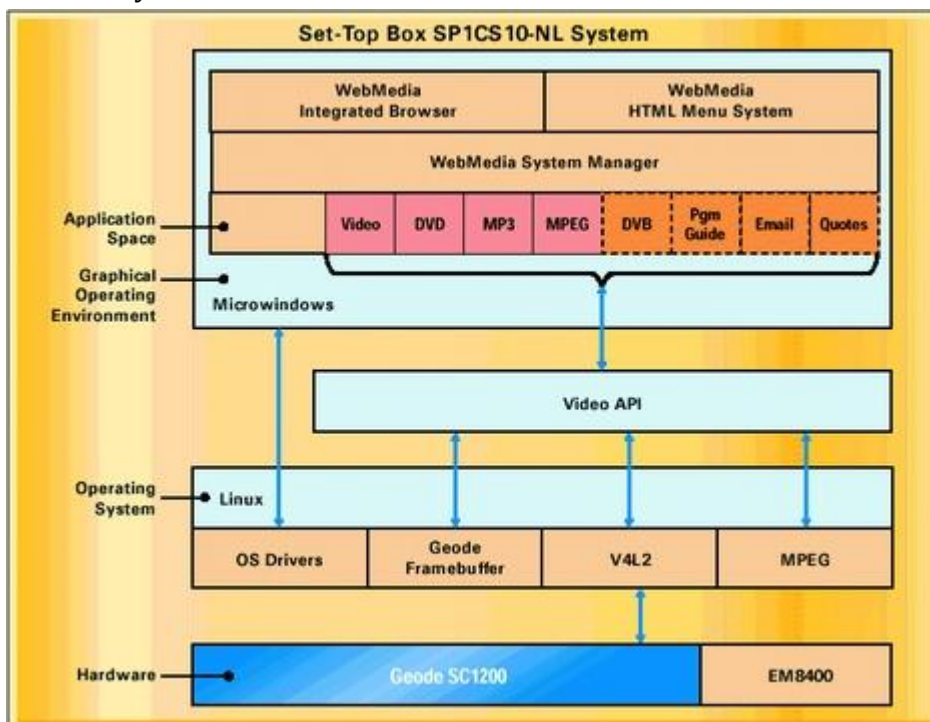


Figure 3. Linux4.TV Software Architecture

a. Graphical Windowing System

Microwindows is used as the graphical windowing system, which easily integrates with the NTSC or PAL-compatible framebuffer provided by the kernel. A nice feature of the set-top box design is that the windowing system is not tied heavily to the hardware

video functions. The windowing system primarily supports WebMedia and the applications layers, which use color keying and Video Middleware API calls in order to interface with video functions. Color keying works by drawing a rectangle in a special color using a standard graphical API, with the Geode's video overlay processor combining the video stream into the framebuffer contents defined by the color-keyed area. Thus, positioning or resizing the video stream is straightforward. WebMedia's use of FLTK allows the X Window System to be used as the windowing system, if desired.

b. Kernel and Device-Driver Layer

The lowest-level software is linked with the kernel as various device drivers and provides low-level chip support for the Geode SC1200 CPU, Sigma Designs MPEG-2 decoder and the Philips SAA7114 NTSC decoder. A number of other drivers are included for the MacPHYTER network interface and stereo audio I/O as well. The kernel is based on version 2.4.3 and uses utilities and the filesystem design from Red Hat 6.2. At boot time, the system performs a normal Linux boot, executes device-driver initialization code, initializes the framebuffer to an NTSC or PAL-compatible screen size and then executes the /etc/rc startup script. There are a number of device drivers that must be available to the Linux operating system in order to provide the level of hardware support needed for the National API. These drivers include audio, video, DVB, DVD, video capture and graphics. In addition, supplementary drivers are needed to provide full support for MPEG, DVD and DVB demultiplexers.

c. Video Middleware Layer

The mid-level layer executes in user mode and implements the device-independent Video Middleware API used to provide a standard interface to the hardware included in the set-top box. The API includes functionality for controlling analog and digital video, selecting the video input, tuner channel selection, DVD, satellite DVB demultiplexing, video blanking interrupt and overlay capabilities. National's Video Middleware uses the Linux Video 4 Linux Two API underneath for interfacing with the kernel drivers. This layer abstracts the operating system and underlying hardware from the application program. Thus, the application programmer need not know which IOCTL for the sound driver sets its frequency; instead, a function call similar to AUD_SetFrequency() is used, which in turn communicates with the audio driver. The details of the driver interface are abstracted from the programmer. Following are descriptions of some of the functional areas of the Video Middleware API.

Audio Subsystem

The audio subsystem API uses an abstract device called an audio player to control the playback of multiple applications while masking the differences of underlying hardware implementations. With audio players, applications are assured of consistent audio playback, regardless of the features or limitations of the target audio device. Audio also includes CD-ROM audio support for playing tracks from an audio CD (a CD-ROM is considered an audio source when an audio CD is inserted).

Video Capture Subsystem

The video capture API enables applications to capture single or multiple frames of live video in real time and save them to a pixel map in graphics memory. Applications can then manipulate the captured images using the functionality of the graphics subsystem or other image-processing facilities.

Overlay Subsystem

The overlay subsystem APIs used in concert with the graphical-windowing system provide arbitration across multiple applications and control which applications are visible. In addition, the overlay subsystem provides a mechanism to overlay graphical information from the graphics subsystem over video.

MPEG Subsystem

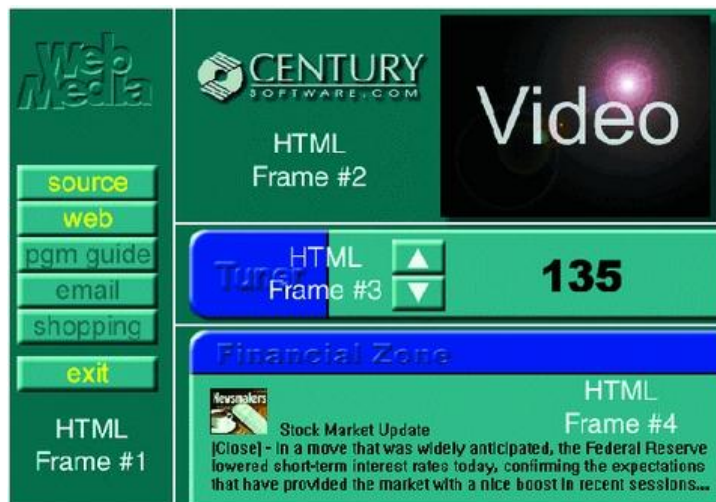
The MPEG transport APIs provide traffic management of MPEG transport streams for digital video. MPEG transport streams are multiprogram data streams of interleaved digital video, audio and data. This API receives MPEG transport streams containing encoded content from the network and extracts the desired program from them. It then separates the audio, video and data components and routes them to the audio decoder, video decoder and CPU RAM, respectively.

TV Subsystem

The TV subsystem API provides a consistent interface to analog and digital broadcast services, so applications developed for one service easily can be extended to work with other services. The TV API also provides functionality that lets end users control interactive TV sets in familiar ways, such as changing the channels and adjusting volume.

d. WebMedia and Applications Layer

The top layer includes all applications software as well as Century's WebMedia, which provides the user interface for the set-top box. WebMedia allows the user interface to be created entirely in HTML, utilizing plugin modules to interface from HTML to the Video Middleware subsystem. The graphical interface is created by providing an HTML representation of the desired display using images combined with an included script to call a plugin module on associated events. Included in WebMedia is an integrated web browser, allowing easy internet access by the set-top box user. Here is a screenshot showing a sample user interface :



-Sample WebMedia HTML-Based User Interface -

WebMedia is a modified version of the ViewML internet browser and uses the FLTK widget set, which allows it to be portable on Microwindows and the X Window System. Because the Video Middleware is separated from the graphical windowing system, it is relatively easy to switch windowing systems. Integrating another browser with the system is straightforward, using the extension mechanism described below.

The entire user interface for the set-top box is written in HTML, which includes both the full-screen viewing and menu-button operations. This allows the set-top box look and feel to be customized easily. Two mechanisms are used to integrate video into the display. The first mechanism involves an extension to the HTML Anchor feature, which allows a specified WebMedia plugin to execute using parameters contained in the anchor tag. For instance, the following HTML causes the DVD plugin to execute a rewind request:

```
<a exec=/bin/wmcmd,dvd,rew><img src=images/rewind.gif>
```

When remote-control arrow keys are pressed, WebMedia outlines successive images associated with anchor tags, allowing the remote control to be used as a pointing device. The prebuilt WebMedia DVD plugin will execute when the remote control is used to select the rewind.gif image. The prebuilt plugins call the Video Middleware API to control the STB hardware functions. Other commands are used to position and scale the video. The second mechanism maps a video stream to the display screen, using the overlay API. A special RGB color is displayed normally on the screen using an HTML image, and color keying is used to merge the video data with the framebuffer contents. This is performed in real time directly by the Geode SC1200 CPU.

In normal operation, the Linux4.TV set-top box runs WebMedia as the controlling application, with HTML specifying a variety of plugins to control the video system integration and user interface. In this manner, the set-top box user interface and total system operation are easily customized for a variety of applications

2. SET-TOP-BOX OPEN SOURCE HARDWARE/SOFTWARE

One of the important remarks that we should consider after the short presentation of the Linux4 Project is that even though the software is free and open source in most of the cases a hardware STB is recommended to have the best results. This is the main reason constructors like Century Embedded Technologies and VT Media Technologies create low cost hardware compatible with Linux embedded software for the STB operating system.

a. Set top box open source de Neuros

- “Open Source Device 2” from Neuros it’s a more conventional box that can encode video at 720p in MPEG-4 format, H.264 D1 resolution (that's DVD quality) from an analog video source, upscale video to 1080i or transcode it for a PMP. It's got a serial port, IRBlaster port, connections for Wi-Fi antenna, USB, LAN, HDMI, composite video and stereo audio, and even packs an SD port. It's shipped with a Linux-based firmware stack, but is completely open to user OS hacking.
- Link is a media extender capable to recover video and streaming from various TV sources (Hulu, NBC, ABC, etc.) and also local audio and video content from USB.

Link’s software is open source, well documented and developed enough so that you can watch video on streaming easily using Neuros.TV interface. This interface is unique so that the user won’t have to browse several web pages to find videos online.

b. VT900 Set-Top Box (STB): Century Embedded Technologies and VT Media Technologies

VT900: offers multimedia services delivered over standard broadband networks, allowing access to services such as interactive television (iTV), Video-on-Demand (VOD), and high speed Internet access. It allows the deployment of a wide range of streamed audio and video content via the Real-Time Streaming Protocol (RTP)

Technical Details

Optimized for National Semiconductor’s Geode system on chip, the VT900 uses Sigma Designs’ EM8400 MPEG Decoder and the MacPHYTER™ Ethernet adapter. Additional peripherals are available and include DVD Player, CDRW, and standard IDE Hardware.

WebMedia supports multiple industry standards including a full featured Internet browser that supports HTML, Javascript, and plug-ins such as Macromedia’s Flash, Secure Socket Layer (SSL) for secure e-commerce transactions, and the TCP/IP protocol. Software upgrades and maintenance are delivered via multicasting to any or all STBs across a subscriber base. This mechanism also ensures that the STB is scalable for future enhancements.

A Software Development Kit is available and offers a suite of software development tools that enable third-party developers to build and customize applications on top of

WebMedia using APIs and application components. The SDK includes documentation, software libraries, sample code, and user interface templates

We have open source software like:

- iSET bundle with support for basic set-top box functionality and several other features that came with the newest versions.
- VT Media Technologies VT900 Developers Kit a complete, integrated, commercial-ready hardware/software bundle based on the low-cost VT900 platform.
- National Semiconductor SP1SC10 Development Platform

For downloading and subscription:
<http://embedded.censoft.com/iset/products/index.php>

c. Boxee Set-Top Box and Remote

Boxee is an open platform set-top box dedicated to the open video platform, along with a special remote. It has interesting features like: Social Network Layer that requires registered user accounts, which form a social network of fellow Boxee users; the „AppBox“ that allows users to download new apps and addons directly from Boxee's GUI; a built-in legal BitTorrent client etc.

Boxee's primary purpose was to streamline video entertainment, but since it's an open platform, it can do a lot more. There's a Boxee app store for downloading third-party apps provided by developers. The main menu has buttons to access music and photos, too. Think of the Boxee as a restriction-free Apple TV.

Software download: <http://www.boxee.tv/download>

d. Review: Logitech's Google TV box

The basic concept: Logitech's box has one key piece of hardware that will keep many current devices from simply porting the Google TV SDK (when it releases in a month or two): and HDMI *input*. In other words, the Revue receives incoming HD content and spits it back to the TV with its own 'meta' content overlaid on it. Examples of this would be watching a movie and having IMDB overlaid on the screen, seeing a commercial and immediately going to the company's website, etc. Google TV is intelligent, connected TV—meta content that is on tap whenever you want to enhance your TV experience.

Google TV, in and of itself, will play a role very similar to the search giant's Android operating system. The software is free, open-source, and its code-base is closely related to the mobile OS. Google provides a level of direction, but hardware makers are what will ultimately drive the software's success. Though several hardware partners are already on board, Logitech's Revue set-top box marks the first big splash in what Google

hopes will be many entries in the market.

e. The YBox2

YBox2 is a DIY networked set-top box that can be used to design customized content to be delivered direct from the Internet.

This project is great for people who want a new platform to experiment with. The video and Internet cores are ready to go and easy to work with. The whole project is open source so you can start hacking your own. If you've ever been curious about the Parallax Propeller chip, the YBox2 is a perky little platform with tons of accessories.

Download software: <http://curl.haxx.se/dlwiz/?type=bin>

f. GeexBox

GeexBox is STB developed under Linux and having the functionalities of a Media Center. It acts like a fully operating system and it can be used as a LiveCD. In our case we tested GeexBox from a USB device. We can easily run movies but once connected to a network the detecting the Network Interface it's an issue even when forcing the OS to do it.

Download software: <http://www.geebox.org/download/>

V. MythTV

1. Introduction

Enter MythTV, a grand unification of personal digital video recording and home theater technology, and a magnum opus of modular design, freedom of expression and personal entertainment. At its core, MythTV is a digital video recording solution composed of several modular components that facilitate time-stretched manipulation of live television feeds, but it's really much more than that. In this rapport, we examine the depth and scope of MythTV's capabilities. We start here from the standard MythTV base, then address the wide-open capabilities that make MythTV more than just a video recording suite - the capabilities that make it into a quintessential home theater PC



(HTPC) system.

MythTV is a software suite that is available as a free download. It consists of a menu system, several plug-ins and a unique frontend/backend network architecture that can be used to transform an existing Linux, BSD, Mac OSX or Windows desktop computer into the multimedia entertainment centerpiece in your den, bedroom or office. Using MythTV, you can morph your dekstop into a video viewfinder, a slideshow gallery, a DVD burning station, a VoIP phone console, a Netflix manager and more.

It has been created by Isaac Richard in 2002 who wanted to use something better than the TV Box which this grow up within is community. Now, he is a strong tool for watching and recording TV.

MythTV was a simple application who were only use for TV and now its more an Home media center, which can handle personnal photos, web video, news and more.

Hardware requirements are easily satisfied: MythTV can run on minimalist EPIA mini-ITX PCs (with or without integrated hardware decoding) all the way up to high-end server hardware with multiple processors and video capture cards. In essence, MythTV has the same minimum requirements as Linux, with the addition of audio/video hardware normally too specialized for typical desktop use. There are a number of compatible capture cards that work with Linux and MythTV.

MythTV consists of a suite of GPL-licensed applications designed to convert a

middling system into a fully-fledged mythical convergence box complete with all of those capabilities one might wish for from a media center, including some features no other such systems possess. For newcomers to MythTV, installation and setup is much easier than in years gone by, as older more experienced users can confirm. Better packaging of MythTV components is now available, and years of development improvements have eased implementation for both experienced and inexperienced users of Linux and MythTV.

On the surface, MythTV provides all the features you would expect from a digital video recording solution: pause, fast-forward and rewind live television, with support for multiple tuner cards (thus, multiple simultaneous recordings) and support for native RTjpeg, MJPEG and MPEG-2 real-time compression for better integration with popular hardware decoders. MythTV can also process Digital Video Broadcast (DVB) data from compliant cards and pc HDTV tuner cards where applicable. An on-screen display provides rich contextual meta-data when browsing channels, also includes key bindings for quick interactive keyboard input and allows for changing and importing of global menu themes and theme-able semi-transparent on-screen displays.

Beyond its glossy surface, MythTV also contains functionality for automatic commercial detection and removal, picture-in-picture support for applicable tuner cards, comprehensive recording scheduling with built-in conflict resolution and a wealth of modules that can enhance any personal entertainment experience. Several auxiliary features come from various modularized components: DVD video and CD audio playback/ripping support, weather forecast information, ROM game emulations, slideshow gallery rotation and electronic program guide support primarily for use with XMLTV-based program guide feeds such as DataDirect.

MythTV is also scalable in the sense that one system may contain multiple tuner cards, or multiple systems can contain one or more tuner cards, where all of them work in concert from a single unified recording schedule to optimize recording capabilities. This is made possible through a well-designed, distributed network architecture that compartmentalizes functionality according to media handling roles: MythTV consists primarily of a single back-end server and one or more front-end clients. Here, a single server can serve double duty as both a front-end client and a back-end master or slave media storage device.

Additionally, the latest MythTV version (0.24, available through subversion revision control or SVN) is better designed to draw menus based. It uses the OpenGL graphics library, contains improved DVD player support and menus, and comes equipped to handle hot-plug removable media and universal plug-and-play (UPnP) devices for introducing new audio and video content.

Main function

- Watch and record TV including TVHD
- Allow timeshifting while watching TV
- Completely automatic commercial detection/skipping
- Intelligently schedules recordings to avoid conflicts
- Parental controls
- Watch and archive DVDs
- Listen to your digital music collection

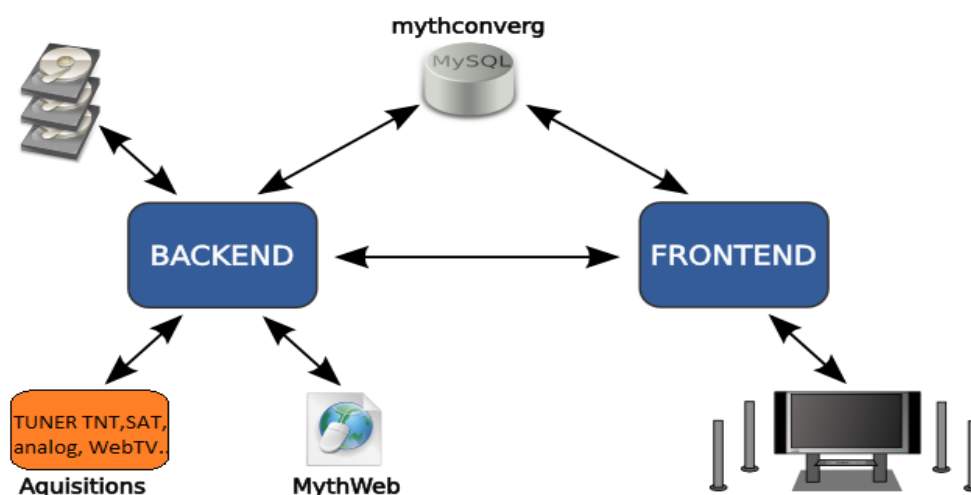
- Schedule and administer many functions remotely via a web browser
- Flexible client/server architecture allows multiple frontend client machines to access content served by one or more backend servers
- RSS support
- Launch your video games
- and many more...

MythTV have a flexible architecture server / client allowing multiple machines to access to the content of one or more servers.

MythTV was design initialy for Linux, BSD and MacOS X but a Windows version exist.

2. MythTV architecture

MythTV need different component to work. The three principal component are MythBackend, MythFrontend and a database. Those component can be installed in a single or several others machines. In fact, it can be split into multiple machine. Below the architecture of MythTV which show how the component dialog each others.



- **MythBackend**
The backend process (mythbackend) is the portion of the system that handles the Video capture cards as well as scheduling recordings on those cards, commercial detection, and transcoding. The backend process interacts with the database primarily.
- **Database**
MythTV stores all of his data into this mysql database; his setup, IP address of the frontend, recording, etc...
- **MythFrontend**
The frontend process (mythfrontend) is responsible for anything that requires interaction such watching recorded programmes or watching TV. It provides the GUI you use on your TV or monitor as well as the LiRc, keyboard, or other input handling for Frontend Navigation. In addition, any MythTV Modules, or plugins, that are installed execute on the same machine that the user is interacting with. As such, the hardware

requirements for a frontend are those necessary to support these features. Underpowered machines may not have enough processing power to decode recordings in software while the same machine could perform these functions if hardware decoding is available.

3. Hardware

Convergence technology plays a defining role in everyday digital appliances we use at home and on the move. It's nice when something just works, when an iPod synchronizes seamlessly with desktop computer data or when an incoming VoIP call registers on your television screen while you're watching a favorite program. In fact, we think you would do well to treat the MythTV suite of software applications as a harness for exercising greater control over those appliances that govern our daily activities outside the workplace.

In and of itself, MythTV represents a framework for building an ultimate home theater system that combines radio, telephony, television and removable/recordable storage media in various useful ways. That said, MythTV is purely software, and this lacks the hardware needed to exercise its capabilities. In other words, MythTV is not something you purchase off the shelf as a finished product like a TiVo unit - but by the same token, that is just another facet of MythTV's unmatched might in the arena of personal digital recording appliances.

As with a Media Center PC you might build yourself, you must supply your own components for a MythTV system, and therefore must forge your own experience based on the budget and usage constraints that factor into your build. Since you won't pay an annual subscription fee for user services related to MythTV operation (as you would with a TiVo) or upfront licensing costs (as with Windows MCE), total cost of MythTV ownership is limited to whatever hardware is already available for the job and that which remains to be purchased - all involving one-time fees.

The following screenshot shows the searchable PVR MythTV hardware database listing provided here : <http://pvrhw.goldfish.org/tiki-pvrhwdb.php> .

System ID	OS/Software	CPU	CPU Type	Memory	H/D	HD Type	Tuner	Audio	Audio	Rating
HomeMedia	Freebox	Intel Celeron	3.15GHz	256MB	Seagate	40GB 7200 RPM	LifeView 2000	nvidia GeForce2 MX400	Onboard	8
Frontend only upstairs machine	MythTV	AMD Athlon	1917MHz Palomino	512MB	Master	80GB ATA-133 150 used in 7200RPM 2MB cache	none - PVR 130 used in 7200RPM server	ATI Radeon 9600 (Farfalla)	external 3D audio card	10
Gen2 mythtv ATI -> Nvidia + PVR150	MythTV	AMD Athlon	1917MHz Palomino	512MB	Master	ATA133 140GB 8MB 150 30GB ATA-100	Happauge PVR-350	nvidia GeForceFX 5200	Sound Blaster Live Value	9
All ATI frontend	MythTV	AMD Athlon	1917MHz Palomino	512MB	Master	80GB ATA-133 7200RPM (cache)	ATI TV Wonder Pro 7200RPM (cache)	ATI Radeon 9600 Pro	Sound Blaster Live Value	8
My first MythTV boxen	MythTV	AMD Athlon	1917MHz	512MB	Seagate	250GB ATA-100 7200 RPM	Happauge PVR-350	nvidia GeForce 5200 FE	onboard	10
MythTV HTPC	MythTV	AMD K2	4400+	2 GB	Seagate SATA II		Happauge PVR-150MC2	nvidia 7800GT	Soundblaster 5 Platinum	5
tyr with xen-vm	MythTV	AMD Athlon	1917MHz	1GB	Master	80GB ATA-133 7200 RPM x2	Happauge WinTV Radio	none	onboard WForce2	8
myth	MythTV	AMD Athlon	1917MHz	512MB	Other	140GB 2x Nova T		nvidia GeForce4 MX400	Onboard Digital - out	9
Home PVE	Freebox	AMD Athlon	1917MHz	1.5GB	Master	40GB ATA100, 200GB, ATA100	Happauge PVR-350	ATI Radeon 9600	onboard	9
multimediasbox	MythTV	Intel Pentium	3.0GHz	512MB	Master	80GB 7200 RPM	Happauge PVR-350	nvidia GeForce 5200	onboard Intel AC97	8

That said, what goes into a MythTV box (and subsequently, what comes out of it) is based entirely on personal choice. You control what does or doesn't qualify for the final build, you tailor the hardware and software components to your liking and you retain ultimate control over its appearance, operation and performance.

So it's more based on a personal choice the setup you should use for your MythTV Box.

In our project we used our laptop, but they are way to powerfull for this applications, so in fact you can use more cheaper hardware to fully function MythTV at a lower price.

4. Detailed Comparaison Chart



There are other Linux-based media applications that can produce a similar media experience, and repurpose standard-fare computer equipment as video recording and

playback gear. Freevo and GeexBox are two likely examples, but they fail to match the feature sets for either MCE or MythTV. The following table outlines some of the prominent differences between the latter pair of products.

MythTV	MCE 2005
Open source; free to obtain, use, and modify	Proprietary; pay to obtain and use but not modify
Native, interchangeable codecs (both OSS and proprietary)	Proprietary codecs acquired separately
Support for MPEG2 transcoding	Third-party codecs provide MPEG2
Setup requires moderate Linux know-how	Simple setup and configuration
Frontends for Linux, BSD, Mac OS X, feeds to Windows	Purely Windows with Xbox frontends and various extenders
Ultra-low system requirements	Moderate system requirements
Support for freely obtainable native and third-party plug-ins	Support for free and variably priced third-party plug-ins
Scalable network architecture (masters/slaves)	Basic network architecture (MCE and extenders)

Perhaps the most profound and telling advantage to MythTV is that its status as a community-based product means anyone can lend a hand in the development process. Possible contributions range from suggesting new and improved features, to creating and implementing new components. In fact, that is how many of the bundled plug-ins and add-ons found their way into the existing MythTV suite, and many more follow suit in each new release.

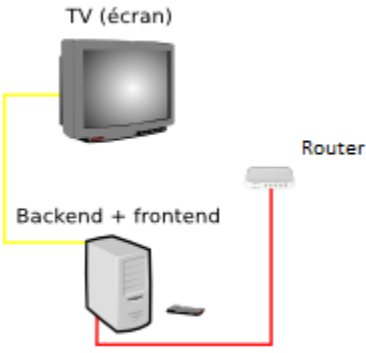
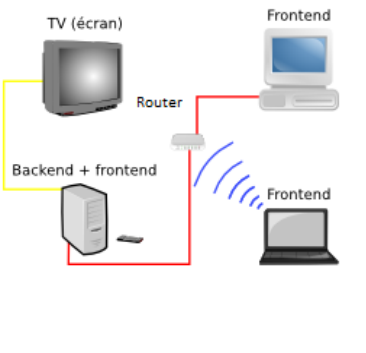
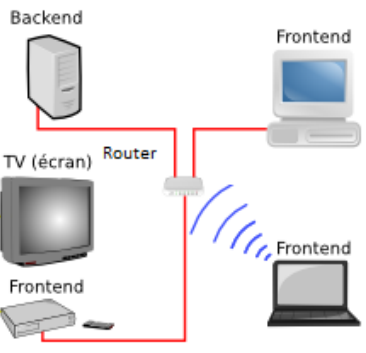
In the next article, we take a look at what truly distinguishes MythTV from all other media system software: its impressive scalable network architecture enables a staggering variety of topologies and arrangements that should have you dusting off and repurposing old systems just for the joy of putting them back to work.

5. Installation

a. Before installation

Before the installation you should determine which configuration are you willing to use, this configuration will determine the level of complexity of the installation. Here we will show only the configuration 1 installation.

Configuration 1	Configuration 2	Configuration 3
-----------------	-----------------	-----------------

		
<p>A machine which is placed under the TV and used as frontend & backend</p>	<p>Same as configuration 1 with multiple frontend.</p>	<p>Backend as no screen. Multiple Frontend user.</p>

System requirements

Minimum System Requirements:

- 1.0 GHz x86 or x86_64 Processor
- 192 MB of system memory (RAM)
- 2 GB of disk space (Frontend Role)
- 20 GB of disk space (Backend Role)
- Graphics card capable of 1024x768 resolution
- Supported TV Tuner Card (Backend Role)

Recommended System Requirements:

- 2.0 Ghz x86 or x86_64 Processor
- 2GB MB of system memory (RAM)
- 10 GB disk space (Frontend Role)
- 160 GB+ disk space (Backend Role)
- nVidia 8xxx+ 256MB+ Graphics Card
- Supported TV Tuner Card (Backend Role)

b. From source

First install required packages:

```
sudo apt-get install build-essential libircclient-dev libasound2-dev libdts-dev
libdvnav-dev \
libxv-dev libxxf86vm-dev transcode libmp3lame-dev subversion qt4-dev-tools libqt4-
dev libsamplerate0 \
libxvidcore4 liba52-0.7.4-dev libfame-dev libcdio-dev msttcorefonts libasound2-doc
libmad0-dev \
```

libid3tag0-dev libvorbis-dev libflac-dev libcdaudio-dev libcdparanoia0-dev fftw3-dev
libfaad-dev \
libsmpeg-dev libmp4v2-dev libtag1-dev mysql-server libvisual-0.4-dev libexif-dev
libxvmc-dev \
libxinerama-dev

Get the latest code for MythTV. In general, you should probably choose the stable version. If you don't mind if your MythTV system is non-functional or has serious bugs that may prevent proper operation/result in lost recordings or inability to play back recordings, you may try using the unstable, development code. If using unstable, development code, you should follow the mythtv-commits and mythtv-dev mailing lists.

Then, simply go in each directory and configure and build each component:

```
./configure --enable-proc-opt  
make  
make install
```

c. From package (MyUbuntu)

Mythbuntu is a community supported add-on for Ubuntu focused upon setting up a standalone MythTV based PVR system. It can be used to prepare a standalone system or for integration with an existing MythTV network. Unlike similar projects, Mythbuntu keeps close ties with Ubuntu and all development is given back to Ubuntu. This architecture allows simple conversions from a standard desktop to a Mythbuntu machine and vice versa. The development cycle of Mythbuntu closely follows that of



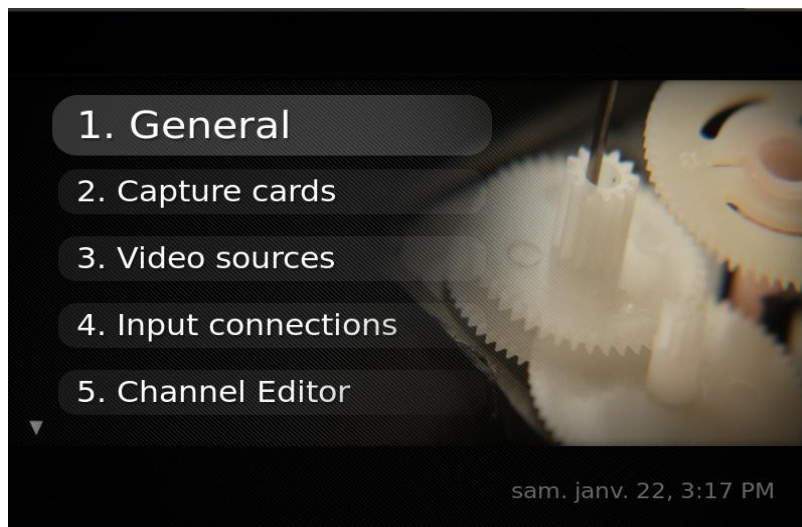
Ubuntu, releasing every six months along side Ubuntu releases.

Download it at Mythbuntu.org and install it like any other Linux distribution. Everything will be setup and ready to go after the installation.

MythTV should open as Ubuntu starts.

d. Configuration

Mythtv-setup allows the configuration for the database access and the TV part. Its also here where we setup the acquisition cards.



This is the first screen of the Backend setup (mythtv-setup)

- General : Configuration of the Backend
- Capture card : Configuration of the capture card
- Video sources : TNT, Analog, Web, etc...
- Input connections : Some cards have different entry, so you should be more specific on which one you want to use.
- Channel Editor : You should have here all the channel you can view and modify there number etc...
- Storage Directories : Where you want to store your recording.

To resume, Mythv stores a « Channel » of a « Video sources » on a « Input connections » of a « Capture cards » and stores the recording into a database.

VI. Experiments

VLC streaming: after having in mind all this concepts and overall idea of the project we adapted the video streaming between server and client into the ADSL network of the lab. The connection through the network works this way:

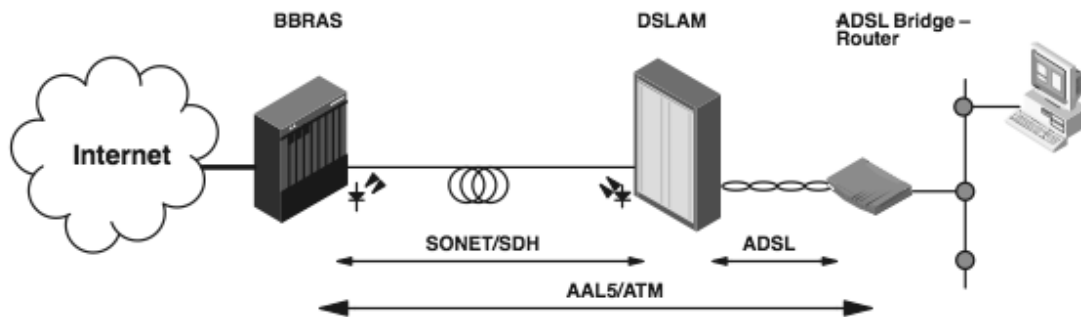
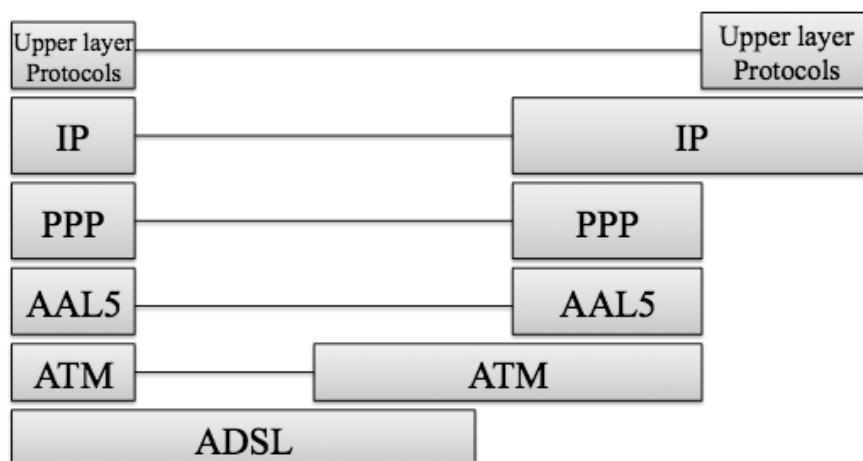


Figure 1. Typical ADSL Overall Architecture

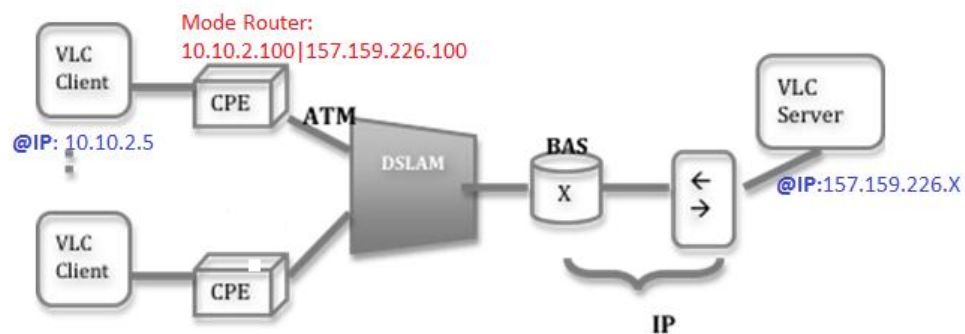
The BAS routes traffic to and from broadband remote access devices such as digital subscriber line access multiplexers (DSLAM) on an Internet service provider's (ISP) network. The BAS aggregates user sessions from the access network. It is here that an ISP can inject policy management and IP Quality of Service.

The DSLAM is deployed in the access network to provide broadband services with ATM via ADSL, SDSL, UDSL, or other xDSL services. The DSLAM incorporates the ATM cross-connect functionality and the central control processor for the entire system.

From a protocol point of view what happens in this architecture with the data is that it generates a large overhead due to different kinds of encapsulations along the path. In all layers and protocols there's generated overhead:



The exact architecture and equipment used in the lab is this:



In a network where a series of routers are needed, IGMP is used to support multicasting, as it is an integral part of IP. To use multicasting, a process on a host must be able to join and leave a group. Group access is identified by the group address and the NIC interface. A host must keep track of the groups that at least one process belongs to and the number of processes that belong to the group. In our case multicast has not worked because there might be a restriction with NAT at the CPE.

At this point we achieved end-to-end connection between VLC client and VLC server through ADSL architecture. By having this connection we streamed a video to the client and by making a request to the IP address of the CPE, the flux is passed onto him.

On the ATM side we have the 10.10.2.0/24 private network. We configure 10.10.2.5 IP address on the VLC client so to be a part of the network. The CPE has the role of a router. It also has the NAT functionality which translates the private IP addresses into public ones so to make the connection with the public network of the VLC server. So we have on the client interface the 10.10.2.100 address which is also the address used to configure the CPE because it is a static one and on the server interface the 157.159.226.100 address connecting to the outside network. The VLC server has an IP address in the 157.159.226.X/24 range, given by the DHCP server.

In the next picture we can see the CPE configuration:

One of the issues encountered at the CPE level is due to the NAT function. As you can see in the following figure the destination port is unreachable (ICMP message type 3) so the CPE does not know where to send the stream:

No. .	Time	Source	Destination	Protocol	Info
88	0.726715	157.159.226.100	PTS 6521.558233333	MPEG	
89	0.735818	157.159.226.100	10.10.2.5	UDP	Source port: 58756 Destination port: search-agent
90	0.744572	157.159.226.100	PTS 6521.571533333	MPEG PES p-frame	
91	0.762085	157.159.226.100	10.10.2.5	UDP	Source port: 58756 Destination port: search-agent
92	0.770748	157.159.226.100	PTS 6521.604900000	MPEG PES p-frame	
93	0.797586	157.159.226.100	PTS 6521.638266666	MPEG PES p-frame	
94	0.805977	157.159.226.100	10.10.2.5	UDP	Source port: 58756 Destination port: search-agent
95	0.815062	157.159.226.100	PTS 6521.651111111	MPEG	
96	0.815123	10.10.2.5	157.159.226.100	ICMP	Destination unreachable (Port unreachable)
97	0.823730	157.159.226.100	10.10.2.5	UDP	Source port: 58756 Destination port: search-agent
98	0.833812	157.159.226.100	PTS 6521.671633333	MPEG PES p-frame	
99	0.841704	157.159.226.100	PTS 6521.674333333	MPEG	

Frame 96 (590 bytes on wire, 590 bytes captured)

Arrival Time: Dec 13, 2010 14:50:07.776188000

[Time delta from previous captured frame: 0.000061000 seconds]

[Time delta from previous displayed frame: 0.000061000 seconds]

[Time since reference or first frame: 0.815123000 seconds]

Frame Number: 96

Frame Length: 590 bytes

Capture Length: 590 bytes

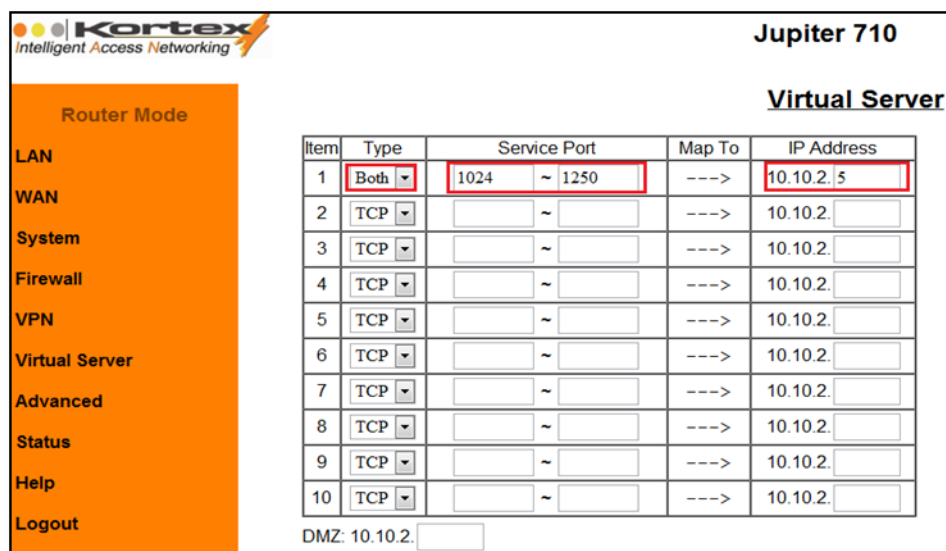
[Frame is marked: False]

[Protocols in frame: eth:ip:icmp:ip:udp:data]

[Coloring Rule Name: ICMP errors]

[Coloring Rule String: icmp.type eq 3 || icmp.type eq 4 || icmp.type eq 5 || icmp.type eq 11]

The solution was to modify the CPE configuration to implement port mapping/forwarding because it receives the streaming through a port but it has to send it to the IP address that demands it (client IP: 10.10.2.5) as you can see in the next figure:



After reconfiguring the CPE we manage to capture the stream at the client level. The following picture shows the case of an UDP unicast stream. For the client the network outside is completely transparent. The flow from the server has as a source address the IP address of the CPE acting as router/NAT. The VLC client has no need to know the VLC server address because it just opens the network streams and receives them if they exist.

No.	Time	Source	Destination	Protocol	Info
1	0.000000	157.159.226.100	10.10.2.5	UDP	Source port: 61353 Destination port: search-agent
2	0.008297	157.159.226.100	PTS 7814.526433333	MPEG	
3	0.016462	DTS 7814.535700000	PTS 7814.549655555	MPEG	
4	0.024819	DTS 7814.569066666	PTS 7814.569066666	MPEG PES	
5	0.032970	157.159.226.100	PTS 7814.572877777	MPEG	
6	0.041544	157.159.226.100	10.10.2.5	UDP	Source port: 61353 Destination port: search-agent
7	0.049731	157.159.226.100	10.10.2.5	UDP	Source port: 61353 Destination port: search-agent
8	0.057824	157.159.226.100	PTS 7814.596100000	MPEG	
9	0.066429	DTS 7814.602433333	PTS 7814.602433333	MPEG PES	
10	0.074624	157.159.226.100	10.10.2.5	UDP	Source port: 61353 Destination port: search-agent
11	0.083103	157.159.226.100	10.10.2.5	UDP	Source port: 61353 Destination port: search-agent
12	0.091353	157.159.226.100	10.10.2.5	UDP	Source port: 61353 Destination port: search-agent

Frame 2 (1358 bytes on wire (1358 bytes captured))

Arrival Time: Dec 13, 2010 15:11:40.626815000

[Time delta from previous captured frame: 0.008297000 seconds]

[Time delta from previous displayed frame: 0.008297000 seconds]

[Time since reference or first frame: 0.008297000 seconds]

Frame Number: 2

Frame Length: 1358 bytes

Capture Length: 1358 bytes

[Frame is marked: False]

[Protocols in frame: eth:ip:udp:mp2t:mpeg-pes]

[Coloring Rule Name: UDP]

[Coloring Rule String: udp]

We can also notice that the UDP packet encapsulates a MP2T packet that represents a Transport Stream used by MPEG encoding method. The TS packets can then be multiplexed and transmitted using broadcasting techniques, such as those used in DVB.

Those TS (MP2T) are formed by Packetized Elementary Stream (MPEG-PES) that are a specification in the MPEG-2 Part 1 (Systems) (ISO/IEC 13818-1) and ITU-T H.222.0. PES defines carrying of elementary streams in packets within MPEG program stream and MPEG transport stream. The elementary stream is packetized by encapsulating sequential data bytes from the elementary stream inside PES packet headers as in the next figure:


```

> Frame 1 (1358 bytes on wire, 1358 bytes captured)
> Ethernet II, Src: BillionE 00:2f:b7 (00:04:ed:00:2f:b7), Dst: Dell_0d:7b:fa (00:26:b9:0d:7b:fa)
> Internet Protocol, Src: 157.159.226.100 (157.159.226.100), Dst: 10.10.2.5 (10.10.2.5)
> User Datagram Protocol, Src Port: 61353 (61353), Dst Port: search-agent (1234)
> ISO/IEC 13818-1 PID=0x45 CC=2
> ISO/IEC 13818-1 PID=0x45 CC=3
> ISO/IEC 13818-1 PID=0x45 CC=4
> ISO/IEC 13818-1 PID=0x45 CC=5
> ISO/IEC 13818-1 PID=0x45 CC=6
> ISO/IEC 13818-1 PID=0x44 CC=12
> ISO/IEC 13818-1 PID=0x45 CC=7

```

As for the ADSL infrastructure the CPE statistic lets us collect some important data as displayed in the next figure:

Kortex Jupiter 710
Intelligent Access Networking

Current Time Fri 21 Jan. 2011, 17:19:54

LAN		
IP Address	10.10.2.100	
Subnet Mask	255.255.255.0	
DHCP Server	enable	
Link Speed	100 Mbps/Full Duplex	
Tx Packets	782/0 (good/bad)	
Rx Packets	1036/0 (good/bad)	
WAN		
PPPoA	connected 0 : 3 : 54	disconnect
Last Error Message		
VPI	0	
VCI	32	
NAT	Enable	
Encapsulation	LLC/SNAP	
IP Address	157.159.226.243	
Gateway Address	157.159.226.241	
DNS Address	157.159.10.12, 0.0.0.0	
ADSL		
Line Status	showtime	
Last Error Message		
Downstream Rate	7616 kbps (interleaved)	
Upstream Rate	800 kbps (interleaved)	
Tx Packets	190/0 (good/bad)	
Rx Packets	88/0 (good/bad)	

We can see that between the CPE and the DSLAM we have an PPPoA (Point to Point over ATM). PPP is used for authentication: user credentials, user's login and password and ATM is used to transport the data encoded into fixed-size cells over virtual circuits (VCs).

The ADSL link between the client and the CPE has a Downstream Rate of 7616kbps and an Upstream Rate of 800kbps. We can easily notice the big rate difference for this asymmetrical line.

We also have the two main functionalities of the CPE that of a Network Address Translation and that of a DHCP server for the private network of the clients' side.

VII. IPTV Multicast

1. Introduction

The most important application of the multicast is IPTV. With the coming of the Triple play including voice, data and video, all in the same network, management of the network resources has become necessary. Therefore multicast is an important part of the IPTV broadcasting.

The main purpose of the multicast is to send data to many people only once, rather than sending a separate copy of the data to each user, routers along the way to the destinations make copies as needed, and multicast receivers just have to request data to receive it.

The destination IP address in a multicast is address of the multicast group,

Typically in IPTV, the TV channels are multicast in the core network and only forwarded in the access network when clients request the respective TV channels. Compared to the core network, the access network has only limited bandwidth capacity involving a good management of traffic in this area.

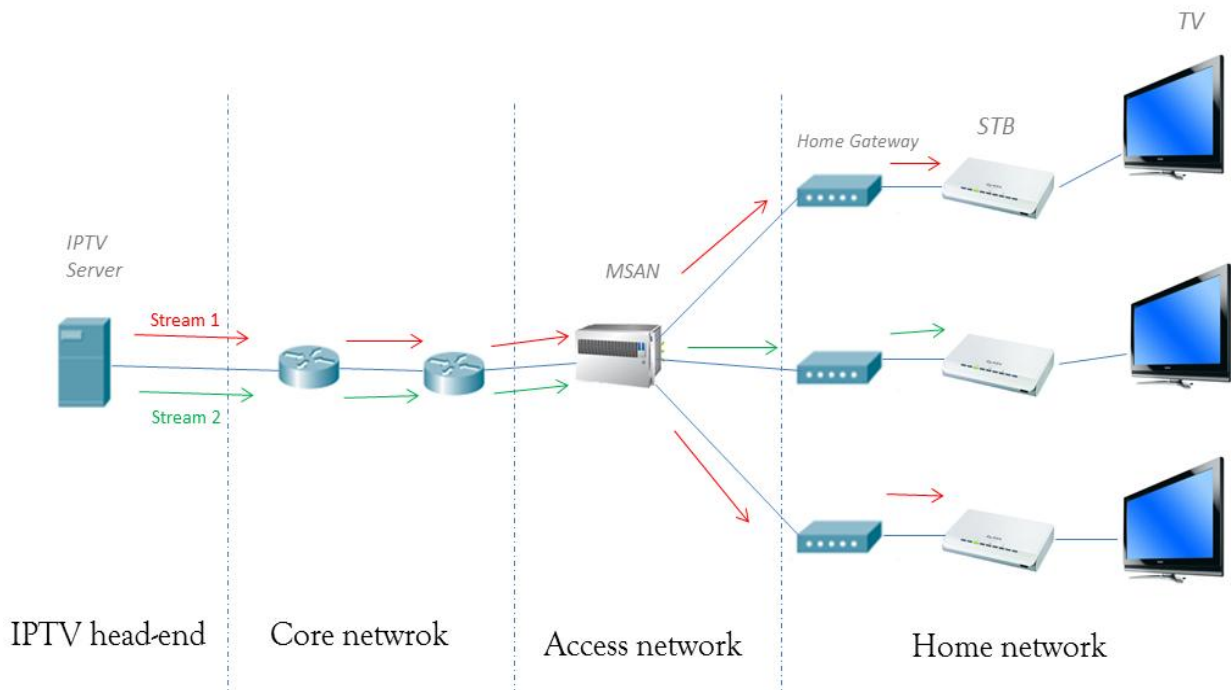


Figure 1: IPTV network distribution

In figure 1, an example of IPTV distribution network for broadcast TV is given. The TV channels are multicast from the streaming server to the Set Top Boxes (STB), the user equipment which decodes the video stream and displays it on a TV. During transport the

stream traverses three networks: the core network, which is maintained by the IPTV service provider; the access network, which connects the user with the service provider and the home network, the network found in the user's premises. The access network and home network are interconnected by a home gateway (HG). The HG is the component which allows devices in the home network, such as a PC or STB, to have connectivity with the outside world. The access network and core network are connected by a Multi Service Access Node (MSAN).

To enable multicast data transport typically two protocols are used: the Protocol Independent Multicast Sparse Mode (PIM-SM) and Internet Group Membership Protocol (IGMP).

2. Multicast routing protocols

Unlike unicast, which routes packets according the destination, multicast routes packets according source.

Thus, the role of the multicast protocol is to build and maintain distributions trees and report the active sources and groups to routers.

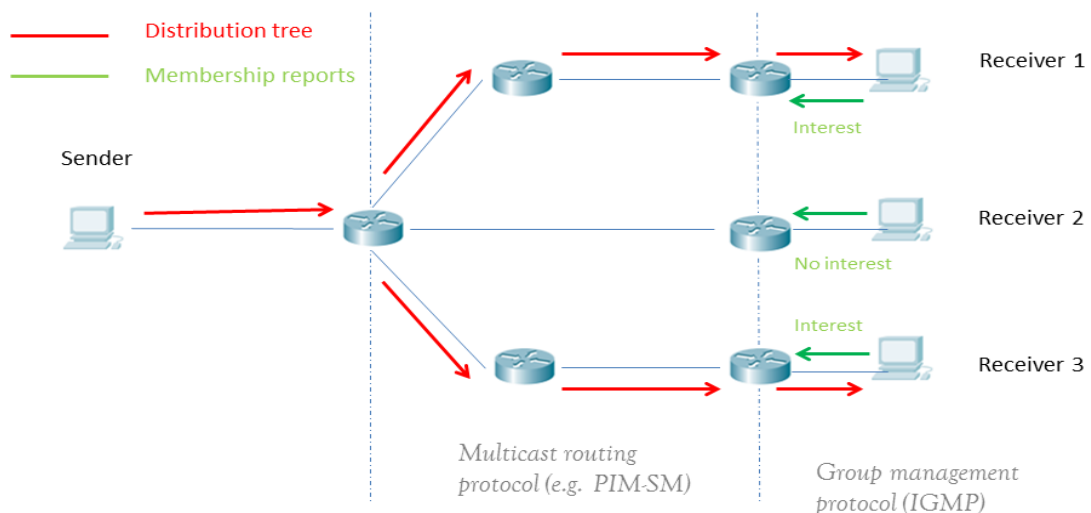


Figure 2: Multicast application

i. PIM-SM (Protocol Independent Multicast Sparse Mode)

PIM-SM is a routing protocol for multicast groups defined by *The Internet Engineering Task Force (IETF)* in the RFC 2117 and RFC 2362 specifications: it allows routers to notify each other of available multicast channels and provides multicast routing functionalities, including the setup of new multicast distribution path from a source to one or more receivers.

It is an independent protocol based on other unicast routing protocols, it is used typically in WAN networks.

Principle:

A router receives explicit Join/Prune messages from those neighboring routers that have downstream group members. The router then forwards data packets addressed to a multicast group, G, only onto those interfaces on which explicit joins have been received.

A Designated Router (DR) sends periodic Join/Prune messages toward a group-specific Rendezvous Point (RP) for each group for which it has active members. Each router along the path toward the RP builds a wildcard (any-source) state for the group and sends Join/Prune messages on toward the RP. The term route entry is used to refer to the state maintained in a router to represent the distribution tree. A route entry may include such fields as the source address, the group address, the incoming interface from which packets are accepted, the list of outgoing interfaces to which packets are sent, timers, flag bits, etc.

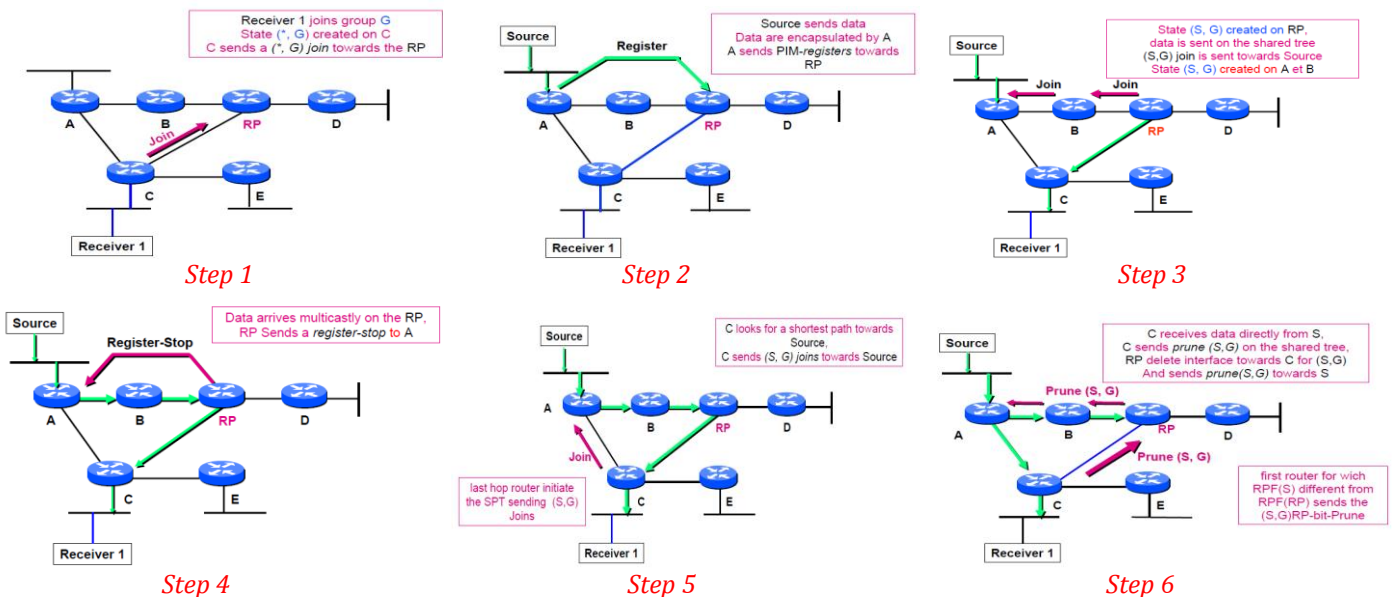


Figure 4: example of PIM-SM scenario

In this scenario, we have illustrated an application for membership of a multicast group using the protocol PIM-SM.

ii. IGMP

IGMP is a subscription protocol with several versions defined also by *The Internet Engineering Task Force (IETF)* in RFC 1112(v1), RFC 3376(v2) and RFC 2236(v3) specifications, which allows clients to subscribe to multicast groups by means of sending membership reports. Access node routers use these IGMP report messages to determine which users are interested in a certain multicast group (TV channel) and thus to determine if packets from a specific multicast group should be forwarded, and to which router ports.

i. IGMP v1

Principle

In each subnet, a router is designed (DR) to send queries every 60 seconds to all the hosts with the multicast address: 224.0.0.1, in which it asks “to which group do you want subscribe?” or “who is subscribed to the multicast group given as parameter”.

Upon receiving a query message, Hosts (the delay timer of whichever expires first) sends an IGMP report to the multicast group address to announce its interest in this group.

Other hosts in the same sub-network suppress themselves report messages for the same multicast group. This mechanism helps reduce traffic over the local subnet.

Hosts wishing to join a group send a report without waiting for request.

But in this version of IGMP, host can't indicate to router that it leaves a group and this is not specified how a DR is designed if there are several routers in the subnet.

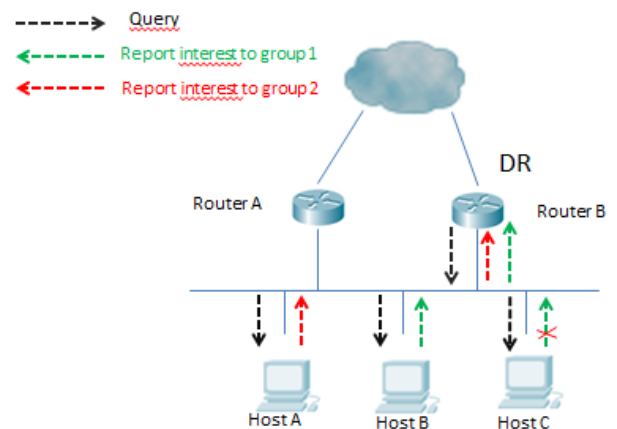
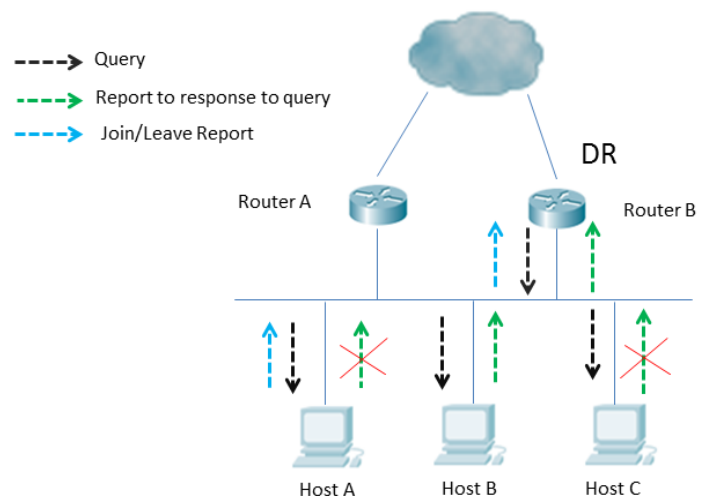


Figure 5: joining multicast groups

a. IGMP v2

This is the most common version of IGMP, it is compatible with version 1 and corrects the limitations of previous version.

Indeed, leaving a group and designed router election mechanism are specified.



Leaving group:

The message to leave a group is directed to the address 224.0.0.2 (All routers). When the router applicant receives this message, it sends a message in response to specific query group left to determine whether there is a host group member in the subnet. If no response is received, the router considers that there are more subscribers to the group.

Figure 6: IGMP v2 diagram

Election DR mechanism:

When a router receives a request from another router, it compares the source IP address with its own. The router whose address is the lowest is selected as the DR on the segment. When a router receives such request higher than its own, he starts a timer of 250 seconds that it stops sending queries. If no message from a DR with a smaller IP is received during this period, requests are sent again.

Nevertheless, this version has some limitations. It is not possible for a host to indicate that it wishes to receive a group from a specific source or exclude a particular source.

b. IGMP v3 and IGMP snooping

Version 3 provides hosts with enhanced control capabilities and provides enhancements of query and report messages. In addition to group-specific queries, IGMP v3 has introduced source filtering modes (Include and Exclude), so that a host not only can join a designated multicast group but also can specify to receive or reject multicast data from a designated multicast source.

The report, whose support is required for versions 1 and 2, is deleted in this version. This facilitates the operation of IGMP snooping and reduces the latency when the last member leaves a group.

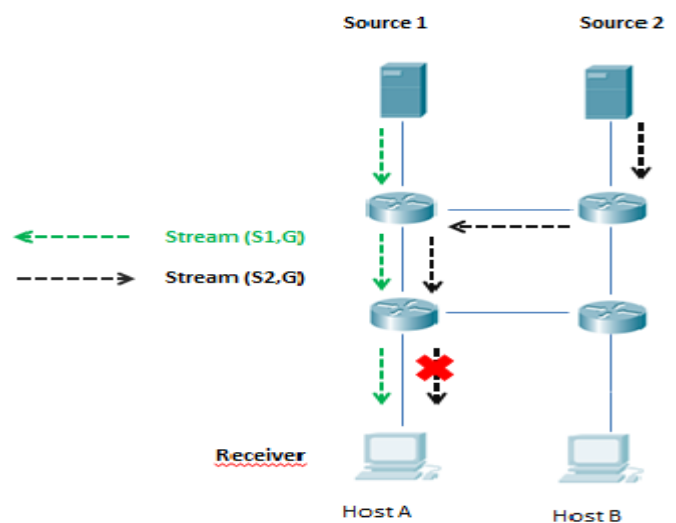


Figure 7: Flow paths of source-and-group-specific multicast traffic

IGMP snooping overview

IGMP snooping is a multicast constraining mechanism that runs on Layer 2 devices to manage and control multicast groups.

By listening to and analyzing IGMP messages, a Layer 2 device running IGMP Snooping establishes mappings between ports and multicast MAC addresses and forwards multicast data based on these mapping.

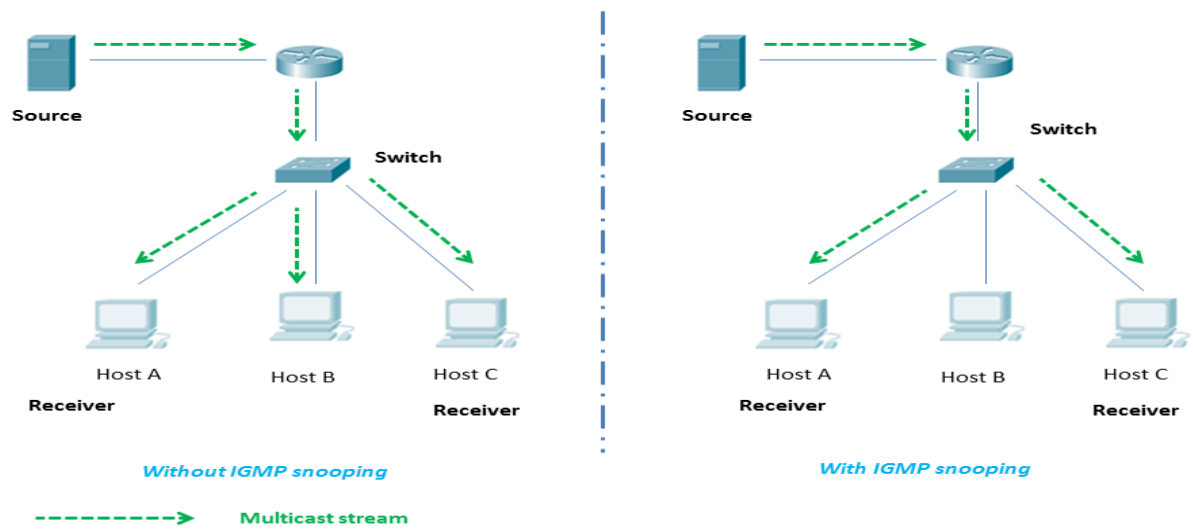


Figure 8: Before and after IGMP Snooping is enabled on the Layer 2 device

VIII. Conclusion

After implementing an IPTV platform and dealing with all the practical troubleshooting that came's along we can say that we have a better understanding of all the issues on video streaming over a network.

First of all, the key fact that we have to keep in mind is that we are working with open source platforms. This brings along advantages and disadvantages.

In the STB case, for instance, it is easy to find the corresponding documentation but not to find coherent and competent answer. It is also difficult to find a virtual STB that has all the functionalities of a real one. In our case the two choices left were GeexBox which has a basic interface and MythTV which is better developed but not yet fully functional. MythTV has also the role of a Media Center, this being the main reason we focused on it. It is important to specify that MythTV can handle streams from DVB sources therefore we manage to receive the TV channels send by Freebox after recovering the channel list in .m3u format. The problem was that we the STB cannot read the .m3u file created locally so we could not recover videos streams from our VLC server. This is an issue for further research and it may be a developer's problem of the software.

Secondly, for the VLC client-server, streaming the UDP/RTP/HTTP posed no problems but the use of the RTSP protocol for VOD was more complex as you could see on the VLC chapter. Using UDP brings less overhead on the link so the latency is lower but does not allow any flow control on the network that's way RTP it's better suited for IPTV.

Thirdly, the ADSL platform helped us understand the two main protocols ATM and IP and how they interconnect and interact when using video streaming. We noticed that the stream rate is not that very high so an all IP network it's more likely to be deployed for HD TV, for example.

In the end we can say that IPTV is a complex field that still needs a lot of research and developing and most of all less ownership and better opening to the outside developers.

Bibliography

http://www.dvb.org/technology/fact_sheets/DVB-T2_Factsheet.pdf

<http://www.opensource.org/>

<http://www.videolan.org/vlc/>

<http://mysettopbox.tv/archive.html>

<http://www.geebox.org>

<http://curl.haxx.se/dlwiz/>

<http://www.mythtv.org>

[Protocol Independent Multicast-Sparse Mode \(PIM-SM\): CFR 2117 Protocol Specification \(Revised\)](#)

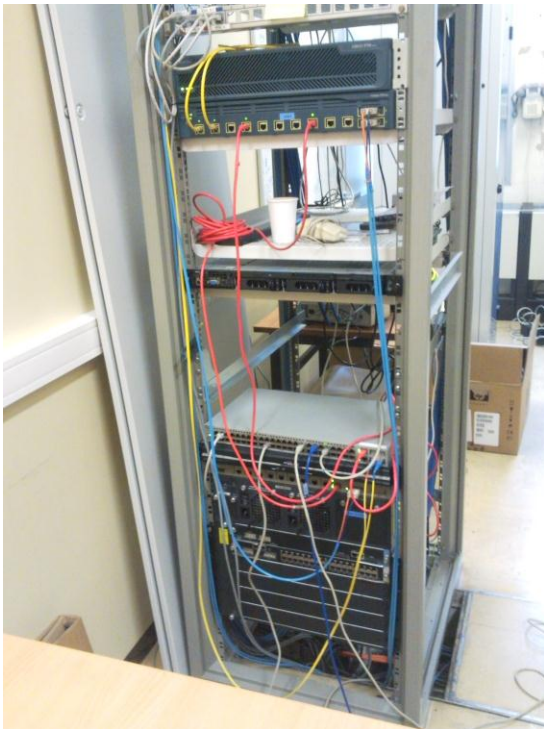
http://www.renater.fr/IMG/pdf/05-PIM_EN_v03.pdf

http://www.h3c.com/portal/Products_Solutions/Technology/IP_Multicast/Technology_Introduction/200702/201192_57_0.htm

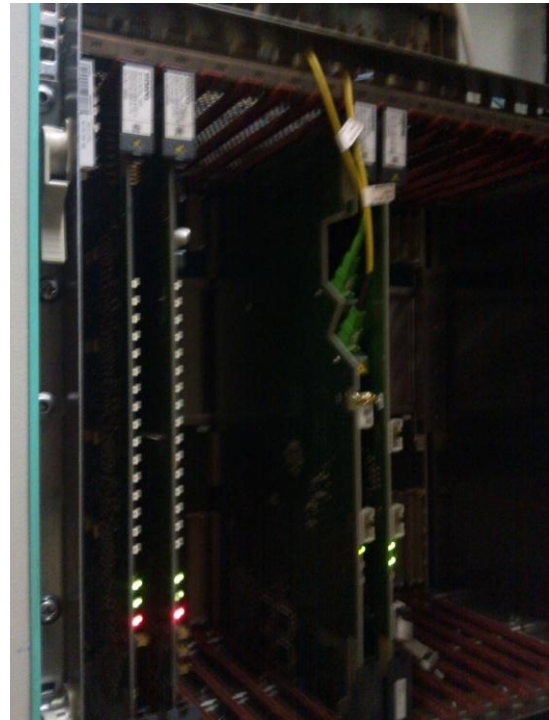
http://fr.wikipedia.org/wiki/Internet_Group_Management_Protocol

Schedulers

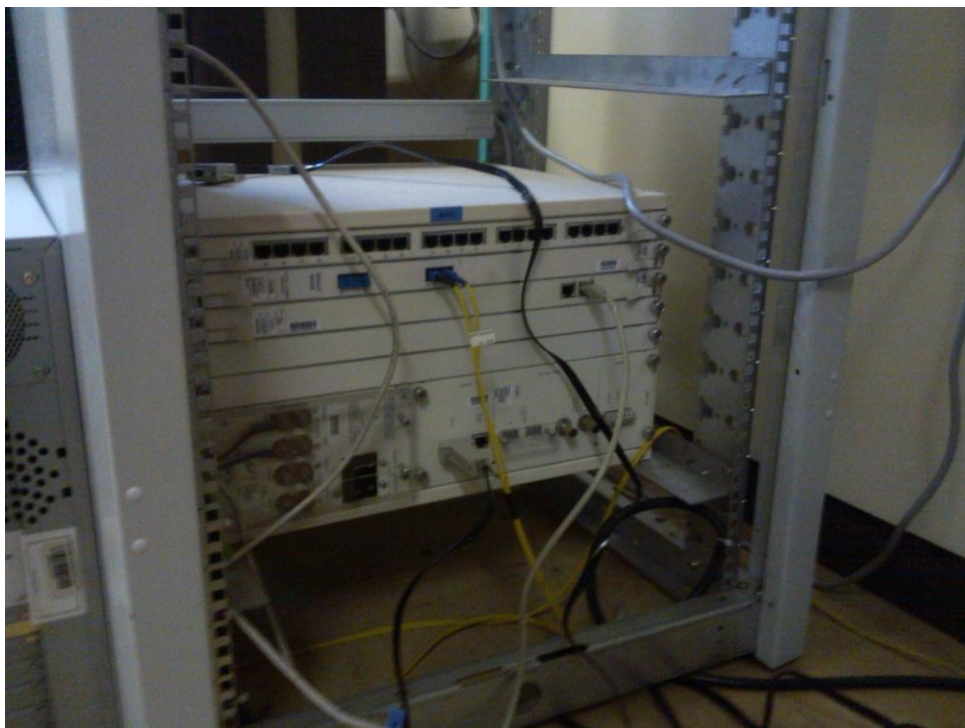
Some photo of the ADSL experiments components:



Switches and routers



ADSL card



BAS (Broadband Access Serve)