

Home Brew IPTV head-end

Thomas Kernen

History & Objectives

- ✦ I wanted to build an IPTV head-end for my home that would suit our viewing requirements
- ✦ It is currently in a “production” state but continues to be upgraded with new hardware and software
- ✦ Word of mouth has led to people being interested in knowing how I did it
- ✦ This presentation is designed to help those who wish to build themselves an IPTV head-end for their home, office, lab, school, campus, etc...
- ✦ Focus of this presentation is on the head-end requirements for streaming live radio and TV services and not Video on Demand (VOD) services
- ✦ All the information in the presentation is provided on an “as is” basis without warranties of any kind, either express or implied

Agenda

- Planing the Head-End
- Building the Head-End
- Media Players
- Support for IPv6
- References

Planning the Head-End

Introduction

- TV viewing habits have evolved from one screen per home with all viewers around it to a multi-screen and multi-viewer environment
- Traditionally homes have had a single physical source for their TV signal. Now there are multiple sources (satellite, cable, terrestrial, etc)
- Receivers are evolving quickly and each generation introduces new and improved features. Hard to keep up and usually requires swapping out gear
- Delinearisation of content means most people want to record and then view it later
- The best off the shelf products tend to provide a subset of the features required to support the statements above, but then are usually locked into their own vendor scheme
- Home brew systems aren't perfect, your own requirements will determine if some or part of what follows is of any use to you

Basics of reception of TV services

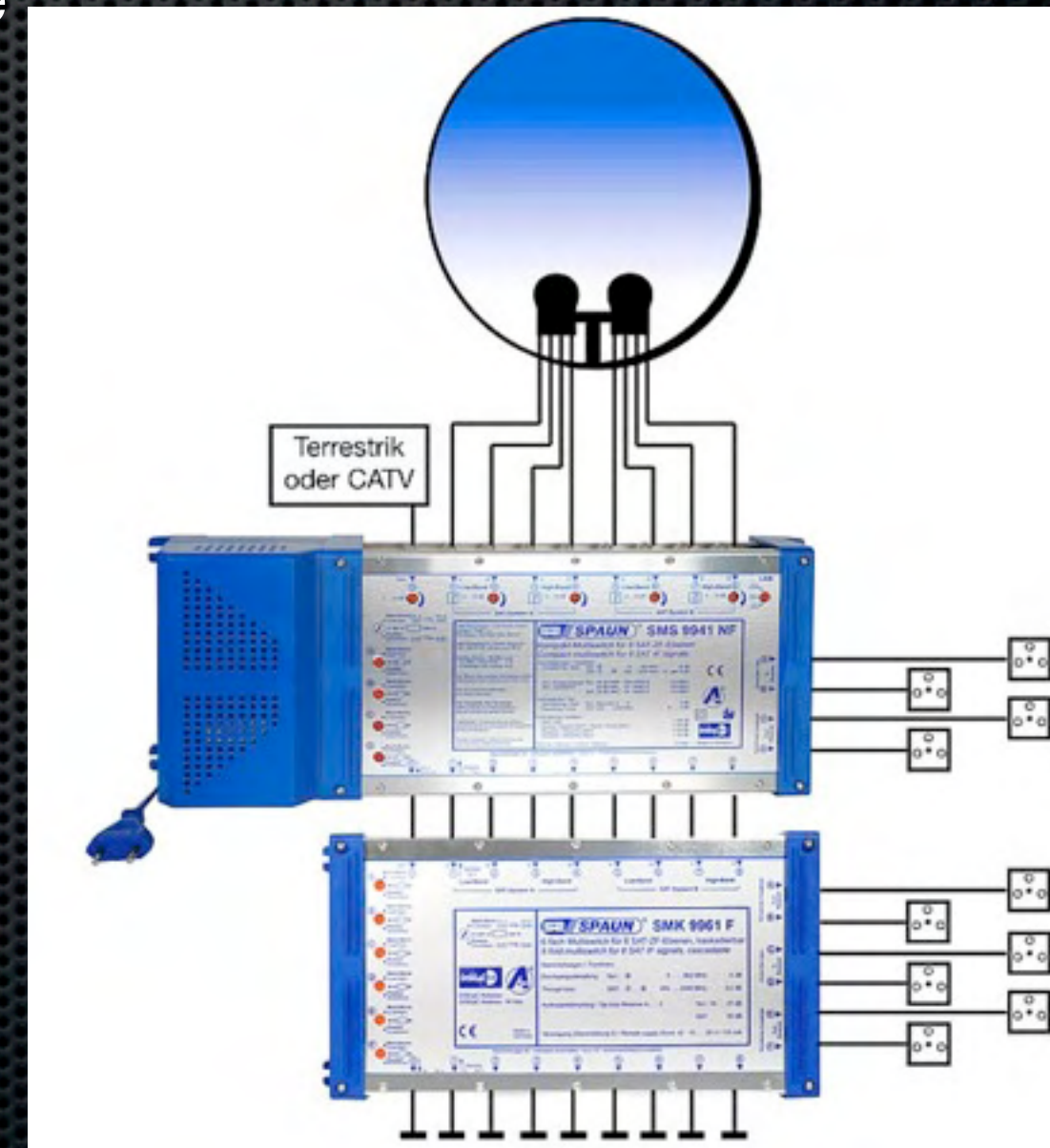
- ✦ In an analogue world, each frequency carries a single service (=TV channel)
- ✦ These days virtually all services are transmitted digitally
 - ✦ “analogue switch off” - ”digital switch over”
- ✦ Each frequency typically carries up to 12 or 13 different digital services
 - ✦ less for HD services
- ✦ These services are modulated on a carrier according to DVB specifications. T and T2 for Terrestrial, S and S2 for satellite, C for cable, ...
- ✦ Each modulation type requires a different receiver
- ✦ All services on a given frequency are contained within an MPEG-2 Transport Stream
 - ✦ Multiple Program Transport Stream (MPTS) or Single Program Transport Stream (SPTS)
- ✦ And may be individually scrambled (= encrypted)

Basics of Reception of TV services

- On a terrestrial or cable source, the full RF bandwidth and therefore services are available at the output of the wall socket. Multiple receivers can be attached without impacting one another
 - To receive a transport stream, the receiver must tune to a specific frequency and setup additional parameters (RF bandwidth, Symbol Rate, modulation scheme, Guard interval, Forward Error Correction, ...)
- With satellite sources, things are more complicated. Different satellites (orbital positions), polarisations (Horizontal/Vertical) and bands (low and high) come into play
 - To receive a transport stream, the receiver must tune to a specific satellite, frequency, polarisation, band and setup additional parameters (Symbol Rate, Forward Error Correction, modulation scheme, ...)
- In a classic receiver these additional parameters are normally hidden from the end user and may be auto detected or preprogrammed

Basics of Reception of TV services

- ✦ To be able to receive multiple satellite feeds we need to include a few lesser known components into the reception system
- ✦ Each Low Noise Block (LNBs) needs to have 4 outputs to provide the 2 polarisations (horizontal/vertical) for each of the 2 bands (low/high). This called a quattro LNB
- ✦ If you need to receive services from 3 different satellites you need 3 LNBs, either mounted on a universal dish that has a bracket for those 3 orbital positions, or on 2 or 3 separate dishes.
- ✦ All the outputs of the LNBs need to be wired to a DiSEqC switch. Different models exist depending on the number of inputs from the LNBs and outputs to the receivers are required. Each satellite will be linked to an input number (usually 1 to 4)



Anatomy of a Transport Stream

- Example of the structure of an MPTS
- Service analysed here is BBC ONE
 - taken from a UK Freeview DVB-T multiplex
 - satellite and cable use the same overall format
- Transport Stream ID: 4100
- Video PID: 600
- Audio PIDs: 601, 602
- Private Data PIDs: 605

The screenshot displays a hierarchical tree view of a transport stream. The root node is 'Transport stream 4100'. Underneath, there are 'Services (6)'. The first service is 'BBC ONE (4.2 Mbps / 25.3%)'. This service contains a '600 MPEG-2 Video (2.4 Mbps / 14.8%)' stream. The video stream details include: PID: 600, Bitrate: 2.439.457 bps / 14.8%, Peak Max: 5.775.703 bps, Max: 2.565.462 bps, Min: 2.256.753 bps, Peak Min: 1.939.495 bps, ES Info (Horizontal Size: 720, Vertical Size: 576, Aspect Ratio: 16:9, Chrominance: 4:2:0, Frame Rate: 25), and Stream ID: 224. Below the video stream are several audio and private data streams: 601 MPEG-1 Audio (262 kbps / 1.6%), 602 MPEG-1 Audio (69 kbps / 0.4%), 605 PES Private Data 1 (4.1 kbps / 0.03%), 650 13818-6 type B (1.10 Mbps / 6.7%), 651 13818-6 type B (150 kbps / 0.9%), 652 13818-6 type B (148 kbps / 0.9%), 1005 13818-6 type B (4.1 kbps / 0.03%), 1006 13818-6 type B (0 bps / 0.00%), and 1007 13818-6 type B (0 bps / 0.00%). At the bottom, there are program and PCR information: Program: 4164, PMT PID: 4164, PCR PID: 600, and Provider: BBC. Other services listed at the bottom are BBC TWO (4.4 Mbps / 26.5%), BBC THREE (7.5 Mbps / 45.5%), and BBC NEWS (3.9 Mbps / 23.8%).

Scrambled services

- There are many scrambling vendors on the market. Different content providers use different systems
- Scrambled services are setup to protect access to the content
- These are usually bound to distribution rights for a specific country
- Some scrambling systems used by content providers may not provide the option of using a standalone subscription card but requires to be paired with a specific provided receiver
- Please check with your content provider(s) if their services can be used with a standard compatible receiver and what scrambling system is used
- Examples (at time of writing):
 - BSkyB (Sky in UK) uses NDS scrambling, paired to the STB
 - BIS Television (in France) uses Viaccess scrambling, no STB pairing

Planning your channel line up

- ✦ Use one of the numerous databases to identify what services you are interested in
- ✦ Make a list of all of them: Frequencies, modulations, scrambling
- ✦ Classify by frequencies, transports and modulations = number of required receivers

Astra 2D (28.2E) - 10773.00 H - Txp: 45 - Beam: Astra 2D												
DVB-S (QPSK) - 22000 5/6 - NID:2 - TID:2045												
Zap	Name	Country	Category	Packages	Encryption(s)	SID	VPID	Audio	PMT	PCR	TXT	Last updated
ZAP	BBC One England 16:9	United Kingdom	General	BBC Sky Digital	Clear	6301	5000	5001 eng 5002 eng	256	5000	5003	2009-07-04 +
ZAP	BBC Two England 16:9	United Kingdom	General	BBC Sky Digital	Clear	6302	5100	5101 eng 5102 eng	258	5100	5103	2009-07-04 +
	ETV			Sky Digital	Clear	6306	5500	5501 eng	260	5500	5503	2009-11-14
	BBC TES 3			Sky Digital	Clear	6315	5200	5201 eng	259	5200	5203	2009-07-04
ZAP	BBC Four 16:9 20h-4h	United Kingdom	Cultural	BBC Sky Digital	Clear	6316	5300	5301 eng 5302 eng	265	5300	5303	2009-07-09 +
ZAP	CBBC 16:9 8h-20h	United Kingdom	Children	BBC Sky Digital	Clear	6317	5200	5201 eng 5202 eng	262	5200	5203	2009-07-04 +
ZAP	CBeebies 16:9 7h-20h	United Kingdom	Children	BBC Sky Digital	Clear	6318	5300	5301 eng 5302 eng	261	5300	5303	2009-07-04 +
ZAP	BBC Three 16:9 20h-5h	United Kingdom	General	BBC Sky Digital	Clear	6319	5200	5201 eng 5202 eng	266	5200	5203	2009-07-09 +
ZAP	BBC One Northern Ireland 16:9	United Kingdom	General	BBC Sky Digital	Clear	6331	5500	5501 eng	257	5500	5503	2009-07-04 +
ZAP	Five 16:9	United Kingdom	General	BBC Sky Digital	Clear	6335	5400	5401 eng 5402 eng	267	5400	5404	2009-08-18 +

Differences between receivers and HEs

- Home receiver tunes to **the** service the user is currently viewing. In a head-end **all** services to be received are permanently tuned
- Home receiver is designed for descrambling **the** service currently viewed. In a head-end **all** services in the transport stream need to be descrambled. This requires special Conditional Access Modules that can descrambled multiple services at the same time. 2 such vendors are Aston (Professional series) and PowerCAM (Pro series).
- Home receiver with IP capabilities will stream **a** channel. In a head-end we stream **all** the services
- To be able to allow any end device in the home to receive any service at any point in time, we need to have multiple receivers tuned to all the channels we wish to be able to view, receiving, descrambling and streaming around the clock

Building the head-end

What you need to build your own HE

▪ Hardware:

- A server case and a motherboard with enough slots for all your receivers (PCI and PCIe)
- Receivers for your different sources (Analogue, DVB-T/T2, DVB-S/S2, DVB-C/C2)
- Conditional Interface (CI) daughterboards for descrambling content protected services
- Conditional Access Modules (CAM) for the daughterboards
- Subscription cards for the different services you wish to descramble

▪ Software:

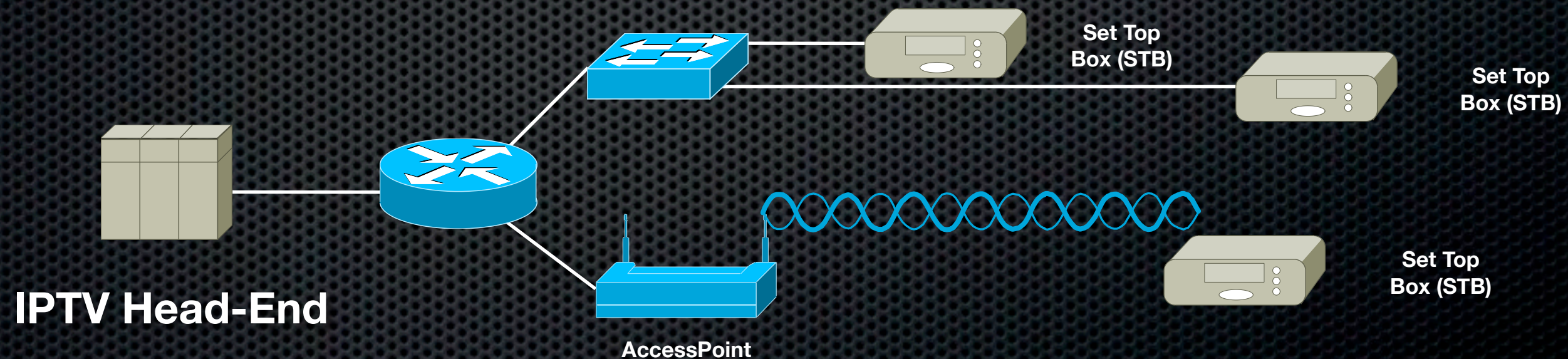
- A modern Linux distribution (Ubuntu for example) with a recent kernel
 - Video4Linux (V4L) DVB drivers and utilities from Linuxtv.org
 - DVB streaming applications such as DVBlaster or MuMuDVB
- A basic understanding of how Linux works and how to get applications running

Hardware

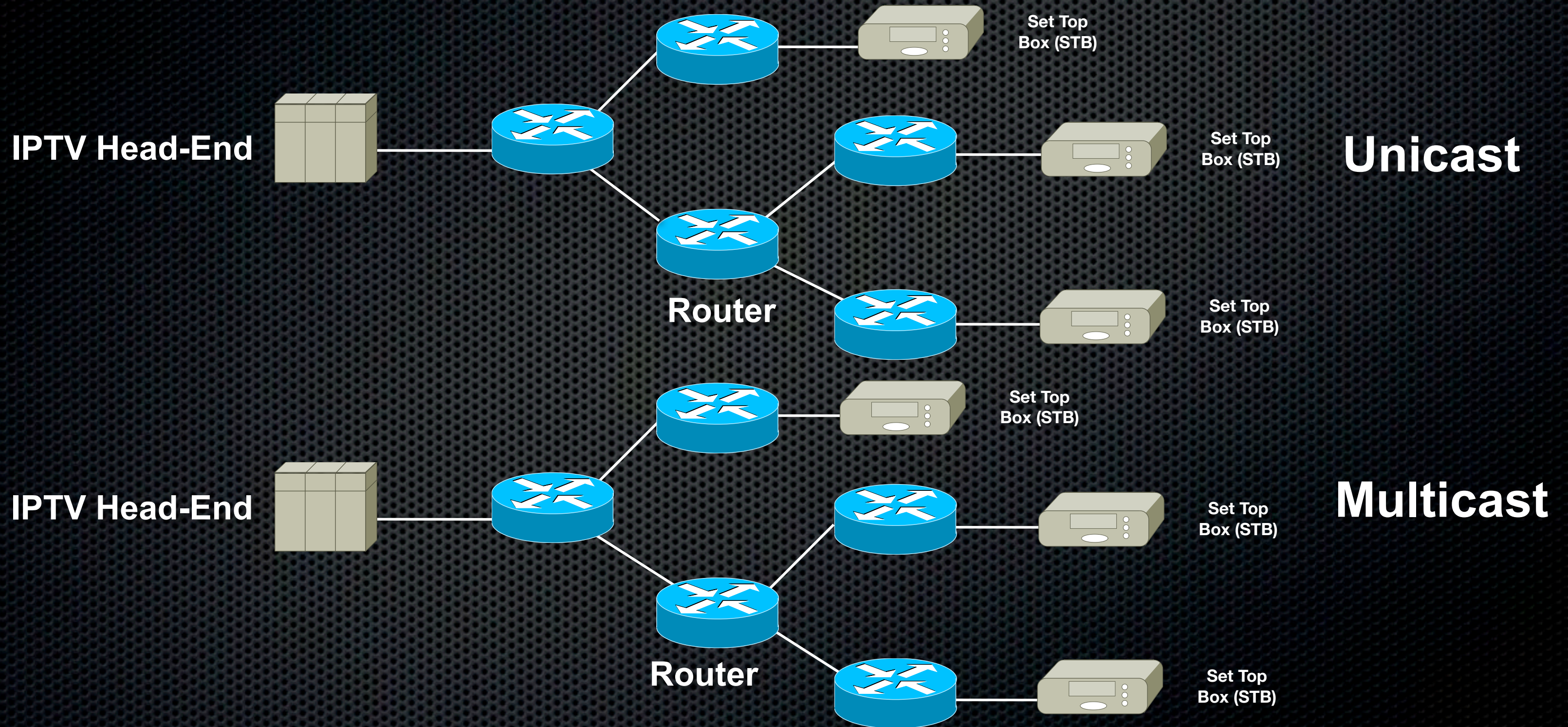
- First you need to find a motherboard with enough PCI and PCIe slots for all the receivers you may need. I've successfully used the Supermicro C2SBC-Q (5x PCI, 2x PCIe)
- To save on PCI/PCIe slots, use dual tuner cards whenever possible
- Receivers with CI onboard to save slots are quite scarce, even more those working under Linux. Dual tuner versions with CI slots onboard are in limited supply with a high price tag
- I have successfully used the following cards and CAMs:
 - Technisat SkyStar II (DVB-S), Technicast Airstar II (DVB-T)
 - Mystique SaTiX (DVB-S2) with CI daughterboard
 - Technotrend S-1500 (DVB-S) with CI daughterboard, T-1500 (DVB-T), S-3200 (DVB-S2)
 - TBS 6980 (DVB-S2) with dual tuner functionality
 - PowerCAM Pro and OneCAM (PowerCam Pro clone)

Networking

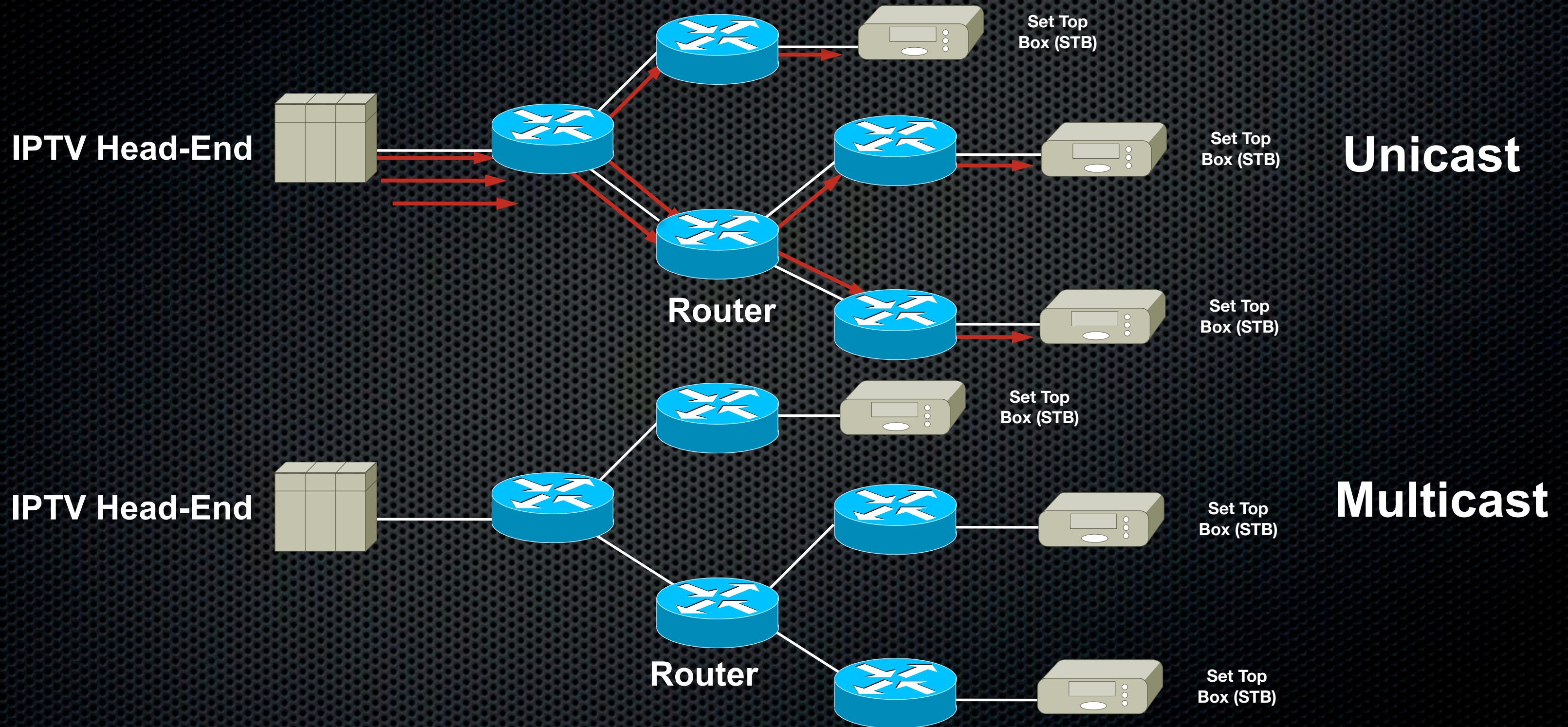
- All networks will differ in one way or another. But chances are that you will have at least a switch, possibly a router, one or more wireless access points. And a variety of end points that could receive services from the head-end
- Note that only the latest 802.11n access points will be able to provide you with decent bandwidth for streaming channels to receivers. They may or may not support multicast



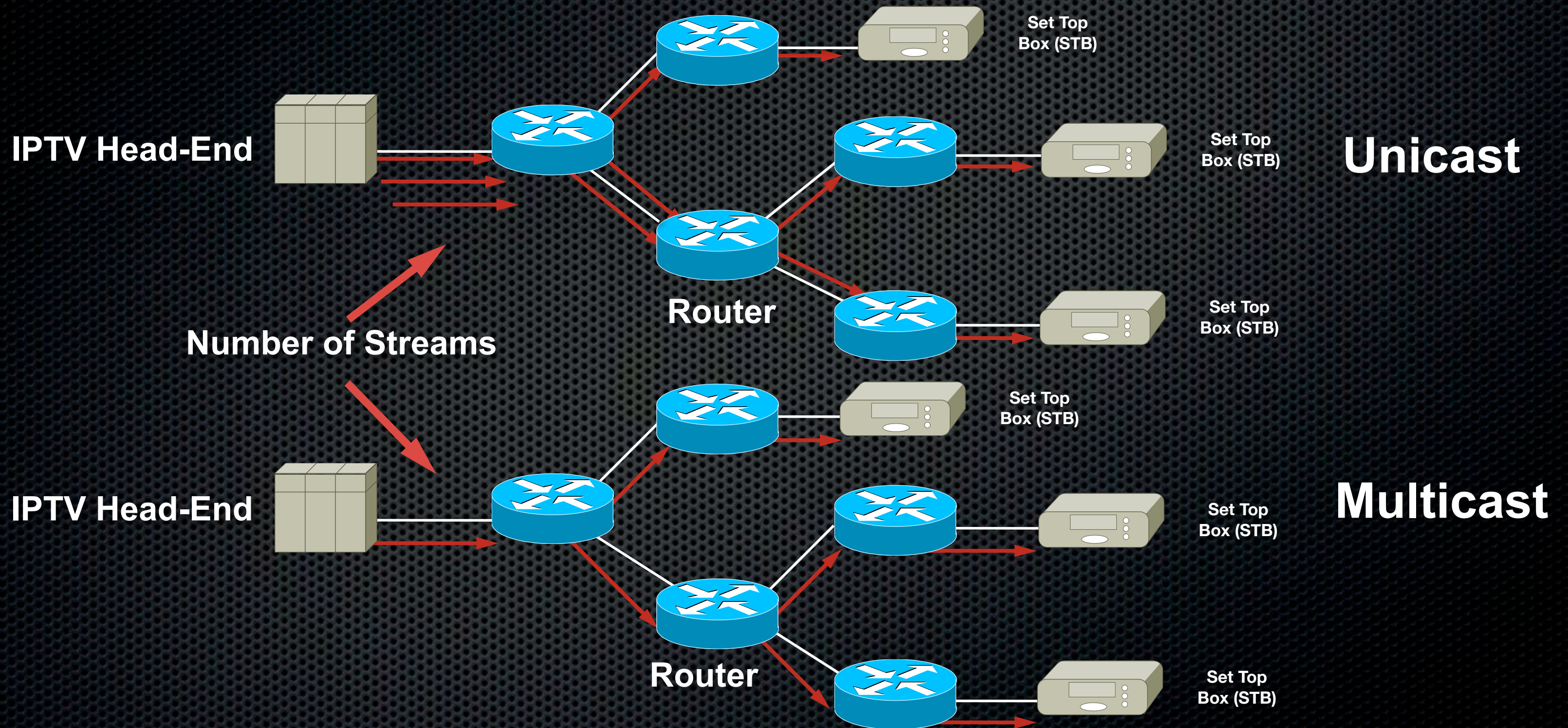
Video streaming: Unicast vs. Multicast



Video streaming: Unicast vs. Multicast



Video streaming: Unicast vs. Multicast



To Multicast or not to Multicast...

- Multicast is the preferred manner to distribute a single source to multiple end devices, but:
 - Many low end (home/small office) routers and switches may not handle multicast properly
 - If there is no routing in your network, make sure you turn IGMP snooping on your switch
 - Source Specific Multicast (SSM) is recommended vs. Any Source Multicast (ASM)
 - The recommended range to be used for your SSM groups is 239.232.x.x/16
 - Most modern operating systems should support IGMPv3 and SSM, excluding Mac OS X
 - The client applications discussed in this presentation support SSM, as long as the OS does too
 - Make sure your hosts have a route for your multicast traffic for 224.0.0.0/4 such as “route add -net 224.0.0.0 netmask 240.0.0.0 dev ethX”
- Unicast will create one stream per receiver, even if 2 receivers are viewing the same service. Whilst this might not be an issue with a couple of receivers, this doesn't scale in an office, school, campus, etc...

Multicast cheat sheet for your router(s)

- This is a very condensed version of what you need to do on your (Cisco) router(s) to allow IP multicast traffic to flow. This may or may not apply to your specific setup and hardware
- Enable multicast globally using the “ip multicast-routing”
- Enable PIM sparse-mode on each interface that sends/receives mcast traffic using “ip pim sparse-mode”
- For IGMPv3/SSM configuration:
 - Add support for IGMPv3 on each interface using “ip igmp version 3”
 - Change SSM range from default to 239.232.x.x using “ip pim ssm range 90” and “access-list 90 permit 239.232.0.0 0.0.255.255”
- For IGMPv2/ASM configuration:
 - Add Rendez-vous Point (RP) using “ip pim rp-address x.x.x.x” where x.x.x.x is the address of the loopback interface of your device. Make sure you add it to all your devices and that it is reachable

Software compilation environment

- In order to be able to download and install the different software packages discussed in the next section you will be the following environment.
- All the syntax used here is specific to Ubuntu, others will need to adapt the syntax.
- **Make** “sudo apt-get install make”
- **GCC** “sudo apt-get install gcc”
- **Kernel-Headers** “sudo apt-get install linux-headers-\$(uname -r) build-essential”
- **Python** is needed for Mercurial “sudo apt-get install python”
- **Git** “sudo apt-get install git-core”
- **Mercurial** “sudo apt-get install mercurial libncurses-dev”

V4L-DVB

- Latest V4L-DVB tree that contains the latest code for your various DVB receiver devices
 - This is important since usually the modules included in kernels is usually out of sync. Many bug fixes
 - To retrieve the v4l-dvb source tree: “hg clone <http://linuxtv.org/hg/v4l-dvb>”
 - To update the sources later on: “cd v4l-dvb” and then “hg pull -u <http://linuxtv.org/hg/v4l-dvb>”
- How to build v4l-dvb kernel modules:
 - Change to v4l-dvb directory “cd v4l-dvb”, build the modules “make”, install the modules “make install”
- dvb-apps is a set of tools to allow you to scan for available channels and then tune to a given service. Useful for preliminary debugging when you can't tune to a given service
 - To retrieve the dvb-apps source tree: “hg clone <http://linuxtv.org/hg/dvb-apps>”
 - To update the sources later on: “cd dvb-apps” and then “hg pull -u <http://linuxtv.org/hg/dvb-apps>”
 - Change into the directory “cd dvb-apps”, build the modules “make”, install “sudo make install”

V4L-DVB modules in the kernel

- [7.870037] saa7146: register extension 'budget_ci_dvb'.
- [7.870109] budget_ci_dvb 0000:06:01.0: PCI INT A -> GSI 74 (level, low) -> IRQ 64
- [7.870236] saa7146: found saa7146 @ mem f82c4000 (revision 1, irq 64) (0x13c2,0x1012).
- [7.870247] saa7146 (0): dma buffer size 192512
- [7.870251] DVB: registering new adapter (TT-Budget-T-CI PCI)
- [7.931683] adapter has MAC addr = 00:d0:5c:04:43:06
- [7.932235] input: Budget-CI_dvb ir receiver saa7146 (0) as /devices/pci0000:00/0000:00:06.0/0000:05:00.0/0000:06:01.0/input/input4
- [8.034327] DVB: registering adapter 0 frontend 0 (Philips TDA10046H DVB-T)...
- [8.012786] budget_ci_dvb 0000:06:02.0: PCI INT A -> GSI 78 (level, low) -> IRQ 65
- [8.012881] saa7146: found saa7146 @ mem f8054000 (revision 1, irq 65) (0x13c2,0x1019).
- [8.012892] saa7146 (1): dma buffer size 192512
- [8.012896] DVB: registering new adapter (TT-Budget S2-3200 PCI)
- [8.070942] adapter has MAC addr = 00:d0:5c:0b:5b:6d
- [8.071348] input: Budget-CI_dvb ir receiver saa7146 (1) as /devices/pci0000:00/0000:00:06.0/0000:05:00.0/0000:06:02.0/input/input5
- [8.414241] stb0899_attach: Attaching STB0899
- [8.443895] stb6100_attach: Attaching STB6100
- [8.454305] LNBx2x attached on addr=8
- [8.454310] DVB: registering adapter 1 frontend 0 (STB0899 Multistandard)...

Checking your devices

- ❖ DVBSnoop can allow you to check if your DVB devices have been detected by the OS
- ❖ DVBSnoop will query the frontend (RF receiver) of your DVB device for card details such as frequency range, capabilities and current parameters
- ❖ This doesn't guarantee the card is in a working state but provides feedback on the detected capabilities and if it is recognised by the OS.
- ❖ Syntax is "dvbsnoop -s feinfo -frontend /dev/dvb/adapterX/frontendX" where adapterX is the dvb device number (starting with 0, 1, 2, ...) and the frontend of that device (0 or 1 for the second tuner of a dual tuner device)

```
dvbsnoop -s feinfo -frontend /dev/dvb/adapter0/frontend0
dvbsnoop V1.4.00 -- http://dvbsnoop.sourceforge.net/
```

```
-----
FrontEnd Info...
-----
```

```
Device: /dev/dvb/adapter0/frontend0
```

```
Basic capabilities:
```

```
Name: "Philips TDA10046H DVB-T"
Frontend-type: OFDM (DVB-T)
Frequency (min): 51000.000 kHz
Frequency (max): 858000.000 kHz
Frequency stepsiz: 166.667 kHz
Frequency tolerance: 0
Symbol rate (min): 0.000000 MSym/s
Symbol rate (max): 0.000000 MSym/s
Symbol rate tolerance: 0 ppm
Notifier delay: 0 ms
```

```
Frontend capabilities:
```

```
auto inversion
FEC 1/2
FEC 2/3
FEC 3/4
FEC 5/6
FEC 7/8
FEC AUTO
QPSK
QAM 16
QAM 64
QAM AUTO
auto transmission mode
auto guard interval
```

```
Current parameters:
```

```
Frequency: 0.000 kHz
Inversion: OFF
Bandwidth: 8 MHz
Stream code rate (hi prio): FEC 1/2
Stream code rate (lo prio): FEC 1/2
Modulation: QPSK
Transmission mode: 2k mode
Guard interval: 1/32
Hierarchy: none
```

Testing your Scrambling modules

- As part of the dvb-apps package, you have a tool called gnutv. This command line application allows you to check your rights for your descrambling cards

```
sudo gnutv -adapter 1 -cammenu
CAM Application type: 01
CAM Application manufacturer: 02ca
CAM Manufacturer code: 3000
CAM Menu string: PowerCam_HD V2.0.4
CAM supports the following ca system ids:
 0x0500
```

```
-----
PowerCam_HD V2.0.4
Select a language
0. Quit menu
1. English
2. French
3. Spanish
4. German
5. Arabic A
6. Arabic B
Select one and press 'OK' to continue
Enter option: 1
```

```
-----
PowerCam_HD V2.0.4
Main Menu
0. Quit menu
1. SmartCard & PIN
2. CAS
3. VP: XXXXX
4. Download Status
5. CSP - Disabled
Select one and press 'OK' to continue
Enter option: 1
```

```
-----
PowerCam_HD V2.0.4
Valid card
0. Quit menu
1. Consultation
2. Authorization
Press 'OK' to continue
Enter option: 1
Please enter your PIN code: XXXX
```

```
-----
PowerCam_HD V2.0.4
Card N?: XXXX
0. Quit menu
1. BIS
2. BIS-PREPAID
3. ViaEutelsat1
4. ViaEutelsat2
5. ViaEutelsat3
6. BIS-EXTRA
Press 'OK' to continue
Enter option: 1
```

```
-----
PowerCam_HD V2.0.4
BIS
0. Quit menu
1. Subscription
2. Pre-booking
Press 'OK' to continue
Enter option: 1
```

```
-----
PowerCam_HD V2.0.4
0. Quit menu
1. Processing...
Please wait ...
Enter option:
```

```
-----
PowerCam_HD V2.0.4
Subscription BIS
0. Quit menu
1. Subscription per class :
2. 11/22/2009 - 12/31/2009 Class:
3. 100,50
4. 09/22/2009 - 10/31/2009 Class:
5. 100,50
6. 09/13/2009 - 09/22/2009 Class:
7. 210,150,103,102,101,100
8. 50,2
9. 08/22/2009 - 09/30/2009 Class:
10. 100,50
```

```
Press 'OK' to continue
Enter option:
Main Menu
0. Quit menu
1. SmartCard & PIN
2. CAS
3. VP: 88730
4. Download Status
5. CSP - Disabled
Select one and press 'OK' to continue
Enter option: 1
```

MuMuDVB

- ✦ MuMuDVB provides RTP/UDP multicast and HTTP unicast streaming capabilities. It takes all services from a transport stream and maps each service to a unique multicast group.
- ✦ By default all services are streamed. Options allow to specify specifically which services are needed (multiple audios, subtitles, Teletext, data, ...)
- ✦ Session Announcement Protocol (SAP) so receivers can listen and build a services listing
- ✦ Download from source: “git clone <http://mumudvb.git.braice.net/mumudvb>”, “cd mumudvb”, “autoreconf -i -f”, “./configure”, “make”, “sudo make install”
- ✦ To launch the application for initial testing (debug) use the following:
 - ✦ “mumudvb -d -vv -s -c <config_file>”
 - ✦ Once you have confirmation the setup is working you can skip the -d value

MuMuDVB

- ✦ As of writing these slides, the latest development builds now offer:
 - ✦ a transcoding feature
 - ✦ Building play lists for VLC Player and MythTV
 - ✦ DVB-T Logical Channel Numbering
 - ✦ Support on OpenWRT platform for mobile setups
 - ✦ Timestamp RTP

MuMuDVB config file (Full Autoconfig)

```
#----- TUNING -----
#The DVB/ATSC card we want to use
card=1
#The Transponder frequency
freq=10773.00
# ---- SATELLITE (DVB-S) ----
#The polarisation (for satellite, can be h, v, l, r)
pol=H
#The symbol rate (satellite)
srate=22000
#LNB Number on DiSEqC switch
sat_number=1
#----- AUTOCONFIGURATION -----
#We want the full autoconfiguration (ie we discover the channels and their pids)
autoconfiguration=2
#Do we want to add the radios to the list of channels (default 0) ?
autoconf_radios=1
#----- NETWORKING -----
#What is the "ip header"?
#The autoconfigured multicast ip have the form header.card.channelnumber
autoconf_ip_header=239.232
#
#Do we want to change the default port (optional) ?
common_port=1234
#
#Do we need to change the default multicast TTL (if you have routers, default
value : 2) ?
multicast_ttl=3
# Adding RTP Headers
rtp_header=1

#----- SAP ANNOUNCES -----
#The sap announces are sent automatically with full autoconfiguration
#Do we want to send the announces ?
sap=1
#
#What is the default playlist group for the SAP announces (optional) ?
sap_default_group=IPTV_Head-End
#Who is the organisation which send the stream (optional) ?
#sap_organisation=my_streamed_tv
#The intervall between the SAP announces in second (default 5)
#sap_interval=10
# ----- PAT REWRITING -----
#If some of the clients are set top boxes we will probably need to rewrite the PAT
pid
rewrite_pat=1
#
# ----- SCRAMBLED CHANNELS -----
#Do we want hardware descrambling ?
cam_support=1
#
#If you have multiple CAMs you can specify the number of the one which will be
used
#cam_number=2
#
#
```

MuMuDVB config file (with PIDs)

```
#----- TUNING -----
#The DVB/ATSC card we want to use
card=1
#The Transponder frequency
freq=10773.00
# ---- SATELLITE (DVB-S) ----
# Select LNB for DiSEqC 0=default 1-4
sat_number=3
#The polarisation (for satellite, can be h, v, l, r)
pol=H
#The symbol rate (satellite)
srate=22000
#----- NETWORKING -----
#Do we want to change the default port (optional) ?
common_port=1234
#
#Do we need to change the default multicast TTL (if you have routers, default
value : 2) ?
multicast_ttl=3
# Add RTP headers
rtp_header=1
#----- SAP ANNOUNCES -----
#The sap announces are sent automatically with full autoconfiguration
#Do we want NOT to send the announces ?
sap=1
#
#What is the default playlist group for the SAP announces (optionnal) ?
sap_default_group=Home
# ----- PAT REWRITING -----
#If some of the clients are set top boxes we will probably need to rewrite the PAT
pid
rewrite_pat=1
```

```
# ----- THE CHANNELS -----
#
#For each channel you have to define :
# * The ip address
# * The port (if you didn't used common_port)
# * The name
# * At least the PMT pid (if there is more than one pid, no autoconfiguration will be
done for this channel)
#
#Ip address
ip=239.232.2.1
#Name of the channel
name=BBC 1 London
#The PIDs (PMT, PCR, Audios, other)
pids=256 5000 5001
#
ip=239.232.2.2
name=BBC 2 England
pids=258 5100 5101
#
ip=239.232.2.3
name=BBC THREE
pids=263 5200 5201
#
ip=239.232.2.4
name=BBC Four
pids=261 5300 5301
#
ip=239.232.2.5
name=Five
pids=262 5400 5401
```

MuMuDVB output

MuMuDVB Version 1.6.1b_20091123

--- Build information ---

Built with CAM support.

Built without transcoding support.

Built with ATSC support.

Built with ATSC long channel names support.

Built with support for DVB API Version 5 (DVB-S2).

Built with pthread support (used for periodic signal strength display, cam support, transcoding, and threaded read).

Originally based on dvbstream 0.6 by (C) Dave Chapman 2001-2004

Released under the GPL.

Latest version available from <http://mumudvb.braice.net/>

Project from the cr@ns (<http://www.crans.org>)

by Brice DUBOST (mumudvb@braice.net)

You decided to send the RTP header (multicast only).

Sap announces will be sent

You have enabled the PAT Rewriting

Full autoconfiguration, we activate SDT rewriting. if you want to deactivate it see the README.

Full autoconfiguration, we activate sorting of the EIT PID. if you want to deactivate it see the README.

The autoconfiguration auto update is enabled. If you want to disable it put "autoconf_pid_update=0" in your config file.

Streaming. Freq 10773000

Using DVB card "ST STV0299 DVB-S"

Tuning DVB-S to Freq: 1023000 kHz, LO frequency 9750000 kHz Pol:H Srate=22000000, LNB number: 1

LNB voltage 18V

DISEQC SETTING SUCCEEDED

FE_STATUS:

FE_STATUS:

FE_HAS_SIGNAL : found something above the noise level

FE_HAS_CARRIER : found a DVB signal

FE_HAS_VITERBI : FEC is stable

FE_HAS_SYNC : found sync bytes

FE_HAS_LOCK : everything's working...

Event: Frequency: 10773452

SymbolRate: 22000000

FEC_inner: 5

Bit error rate: 768

Signal strength: 59838

SNR: 63117

FE_STATUS:

FE_HAS_SIGNAL : found something above the noise level

FE_HAS_CARRIER : found a DVB signal

FE_HAS_VITERBI : FEC is stable

FE_HAS_SYNC : found sync bytes

FE_HAS_LOCK : everything's working...

Card 0 tuned

Autoconfiguration Start

Autoconfiguration done

Diffusion 8 channels

Channel number : 0, name : "BBC 1 NI" TS id 6331

Multicast ip : 239.232.0.0:1234

Channel number : 1, name : "FIVE" TS id 6335

Multicast ip : 239.232.0.1:1234

Channel number : 2, name : "BBC 1 London" TS id 6301

Multicast ip : 239.232.0.2:1234

Channel number : 3, name : "BBC 2 England" TS id 6302

Multicast ip : 239.232.0.3:1234

Channel number : 4, name : "ETV" TS id 6306

Multicast ip : 239.232.0.4:1234

Channel number : 5, name : "BBC TES 3" TS id 6315

Multicast ip : 239.232.0.5:1234

Channel number : 6, name : "CBBC Channel" TS id 6317

Multicast ip : 239.232.0.6:1234

Channel number : 7, name : "CBeebies" TS id 6318

Multicast ip : 239.232.0.7:1234

DVBlast

- DVBlast is a simple and “stripped down” version of VLC. It doesn’t do any kind of processing on the streams, transcoding, PID remapping or remultiplexing
- The focus is on stability and minimal tampering with the stream source
- Designed to open DVB device, tune, place PIDs filters, config CAM and demux to output
- It has one external dependency: libdvbpsi
 - Install libdvbpsi before you compile DVBlast
 - Latest release code here: <http://download.videolan.org/pub/libdvbpsi/0.1.6/libdvbpsi5-0.1.6.tar.bz2>
 - “./configure --enable-release”, then “make”, “make install”
- Download latest release code here: <http://downloads.videolan.org/pub/videolan/dvblast/1.1/dvblast-1.1.tar.bz2>
 - Currently no configuration options, “make” is all that is required. Move bins to your favorite directory

DVBlast

- Command line for tuning parameters, config file for channels to be streamed
- “dvblast -u -f 10773000 -a 0 -n 0 -s 22000000 -v 18 -t 3 -S 1 -e -c <config.cfg>”
 - -u budget mode (no hardware filtering)
 - -f = frequency, a = adapter number, n = frontend, s = symbol rate
 - -v 13|18 voltage to be applied to trigger Vertical|Horizontal polarisation
 - -t = Multicast TTL, -S satellite number (0, 1-4)
 - -e enable EPG pass through (EIT data)
 - -c configuration file containing the specific streams you want to multicast
- Options:
 - For DVB-S2 you must indicate the modulation (QPSK or PSK_8)
 - For DVB-T the bandwidth of the multiplex needs to be added (usually 8MHz)

DVBlast (config file)

- Format is the following: `<IP>[:<port>][/udp] <always on> <SID> [<PID>,*]`
 - There are three ways of configuring the PIDs to stream :
 - 1. SID-based
 - `239.255.0.1:1234 1 10750`
 - DVBlast will stream all known PIDs from service 10750 (video, audio, and subtitles). The resulting stream is fully MPEG-compliant, with PAT and PMT.
 - 2. SID and PIDs
 - `239.255.0.1:1234 1 10750 1234,1235`
 - DVBlast will stream SID 10750, but only PID 1234 and 1235 will be output. Other known PIDs will be discarded and removed from the PMT. The list of PIDs in the config file does not include the PAT and PMT, but it must include the PCR PID if it is different from the video or audio PID, otherwise the stream won't be compliant.
 - 3. PIDs only
 - `239.255.0.1:1234 1 0 0,128,1234,1235`
 - DVBlast will only stream the PIDs passed. No PAT and PMT will be generated, so if they are not included the stream won't be compliant. Also the included PAT and PMT may contain ghost programs or ESeS.

DVBlast output

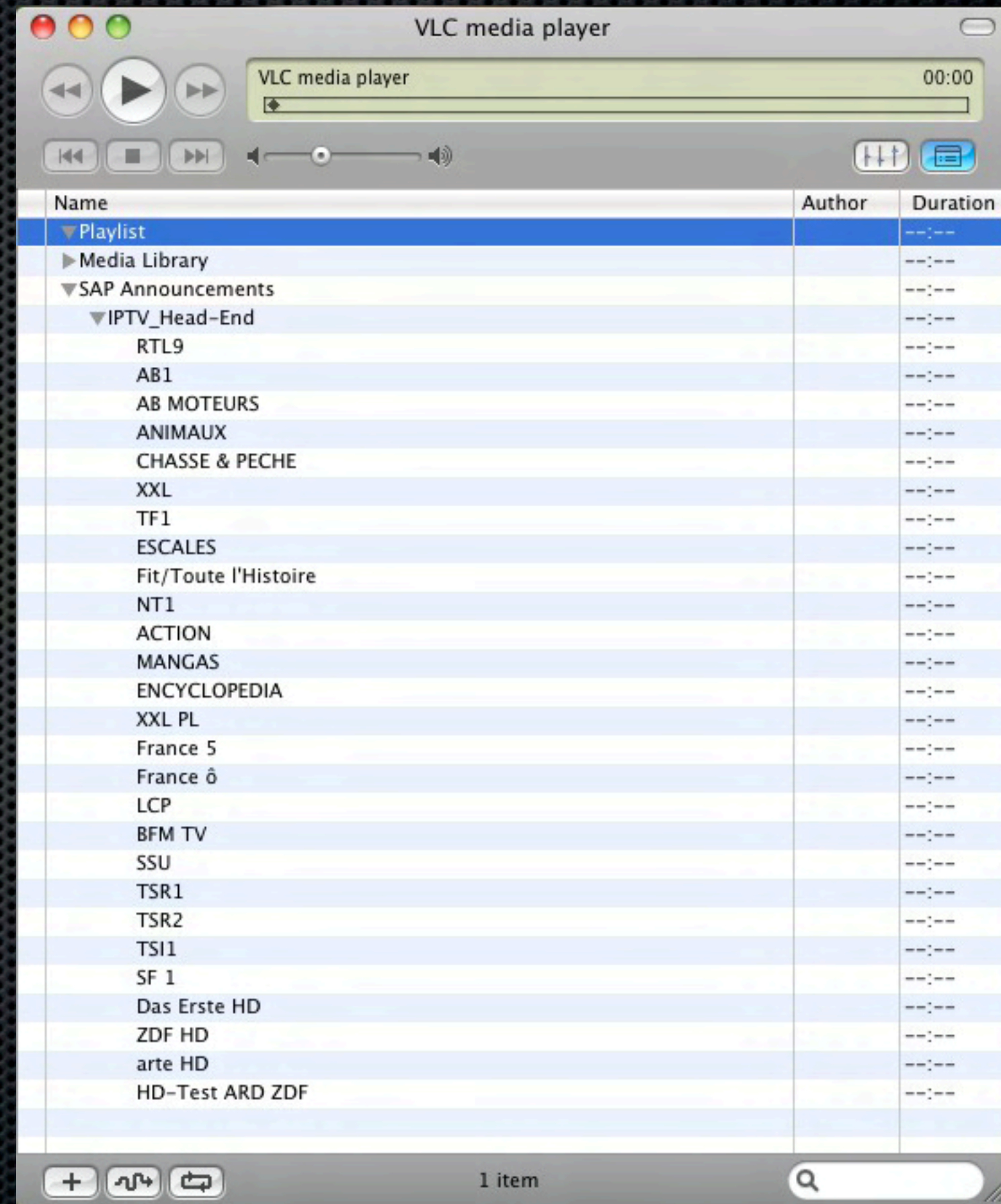
```
sudo ./dvblast -u -f 10773000 -a 0 -n 0 -e -t 3 -v 18 -s 22000000 -S 1 -c BBC.cfg
warning: restarting
debug: using linux-dvb API version 5
debug: frequency 10773000 is in Ku-band (lower)
debug: configuring LNB to v=18 p=0 satnum=1
debug: tuning QPSK frontend to f=10773000 srate=22000000 modulation=legacy
warning: failed opening CAM device /dev/dvb/adapter0/ca0 (No such file or directory)
debug: setting filter on PID 8192
debug: conf: 239.232.232.21:3000 w=1 sid=6301 pids[0]=-1,-1,-1,-1,-1...
debug: conf: 239.232.232.22:3000 w=1 sid=6302 pids[0]=-1,-1,-1,-1,-1...
debug: conf: 239.232.232.23:3000 w=1 sid=6319 pids[0]=-1,-1,-1,-1,-1...
debug: conf: 239.232.232.24:3000 w=1 sid=6316 pids[0]=-1,-1,-1,-1,-1...
debug: conf: 239.232.232.25:3000 w=1 sid=6335 pids[0]=-1,-1,-1,-1,-1...
libdvbpsi error (PSI decoder): TS discontinuity (received 2, expected 0) for PID 18
libdvbpsi error (PSI decoder): TS discontinuity (received 15, expected 0) for PID 17
libdvbpsi error (PSI decoder): TS discontinuity (received 13, expected 0) for PID 0
debug: new PAT ts_id=2045 version=30 current_next=1
[snip]
debug: frontend has acquired stable FEC
debug: frontend has acquired sync
debug: frontend has acquired lock
debug: - Bit error rate: 19712
debug: - Signal strength: 59518
debug: - SNR: 63069
debug: new PMT program number=6301 version=5 pid_pcr=5000
[snip]
```

```
BBC.cfg Configuration file
:BBC 1 London
239.232.232.21:3000      1      6301
:BBC 2 England
239.232.232.22:3000      1      6302
:BBC THREE
239.232.232.23:3000      1      6319
:BBC FOUR
239.232.232.24:3000      1      6316
:FIVE
239.232.232.25:3000      1      6335
```

Media Players

VLC Media Player

- ✦ A well known Open Source media player, supports most OSes, a large number of file formats, audio and video codecs
- ✦ In the options enable support for SAP announcement to pickup the services via the playlist menu (that's if sessions are announced and being received)
- ✦ Performance, specifically with High Definition content is therefore CPU bound
- ✦ If no SAP, you can load a file containing all the services settings
- ✦ GPU acceleration to be introduced in VLC 1.1 for Windows Vista/7 and Linux



VLC Media Player - Services list format

- ✦ Using the Extended M3U Playlist format, you can create a service list for VLC
- ✦ Load the file directly into VLC locally or remotely over your network
- ✦ This will load the services list into the VLC playlist and allow you to select the services you wish to view
- ✦ Double click on the service and your up and running

```
#EXTM3U
#EXTINF:0,1 - Channel Name
rtp://@239.232.1.1:1234
#EXTINF:0,2 - Channel Name
rtp://@239.232.1.2:1234
#EXTINF:0,3 - Channel Name
rtp://@239.232.1.3:1234
#EXTINF:0,4 - Channel Name
rtp://@239.232.1.4:1234
#EXTINF:0,5 - Channel Name
rtp://@239.232.1.5:1234
```

```
#EXTM3U
#EXTINF:0,1 - Channel Name
rtp://192.168.1.100@239.232.1.1:1234
#EXTINF:0,2 - Channel Name
rtp://192.168.1.100@239.232.1.2:1234
#EXTINF:0,3 - Channel Name
rtp://192.168.1.100@239.232.1.3:1234
#EXTINF:0,4 - Channel Name
rtp://192.168.1.100@239.232.1.4:1234
#EXTINF:0,5 - Channel Name
rtp://192.168.1.100@239.232.1.5:1234
```

MythTV

- ✦ MythTV is an Open Source software digital video recorder (DVR). It has been under development since 2002 and contains most features a modern DVR would provide
- ✦ It is split in 2 main components:
 - ✦ MythBackend: Server applications dealing with scheduling, recording, keeping TV listings up to date
 - ✦ MythFrontend: Provides end-user interface and communicates with one or more backends
- ✦ Since version 0.21 and with some extra work you can integrate IPTV multicast sources as a valid input for MythTV
- ✦ The MythTV wiki provides further details on how to do this for some IPTV service providers, so does a link in the reference section. This will require some tweaking to get it running

Support for IPv6

Network support for IPv6 (quick intro)

- In the same way that multicast is not always well (or at all) supported on some networking devices, this applies also to IPv6 multicast
- First select the scope you wish to use for the distribution of your content
 - Important to contain your multicast traffic within specific boundaries
- Enable IPv6 multicast globally using the “ipv6 multicast-routing”
- IPv6 needs to be configured and enabled on all the networked interfaces that will be in use
- PIM Sparse Mode and MLDv2 enabled by default in Cisco IOS (when features are supported)
- For MLDv2/SSM configuration (Supported in Windows Vista/7 and Linux 2.6 kernels)
- For IGMPv2/ASM configuration (Rendez-Vous Point)
 - Add Rendez-vous Point (RP) using “ipv6 pim rp-address x.x.x.x”. Same instructions as IPv4 syntax

Head-end & receiver support for IPv6

- Linux supports IPv6 in the 2.6 kernel branch that is used in the current distributions
- Currently MuMuDVB doesn't support IPv6 but this is planned at some stage
- DVBlast supports IPv6 since r93 commit of the code. This is in the development branch and will therefore be available in the 1.2 release of DVBlast
- No change on the command line syntax, only in the config file syntax for the multicast address. Note that the IPv6 address must be in between brackets
 - Ex: [FF15::ABCD]:1234 1 10750
- VLC Media Player supports IPv6 out of the box
- MythTV supports IPv6 with a bit of extra work

Summary

- Basics related to building a video head-end have be highlighted
- Advantages and constraints related to such a setup were pointed out
- Keep in mind your own requirements: services, network, receivers
- Have fun with your own project, please document and share with others....

References

References

- DVB-T: <http://en.wikipedia.org/wiki/DVB-T> & <http://en.wikipedia.org/wiki/DVB-T2>
- DVB-S2: <http://en.wikipedia.org/wiki/DVB-S2>
- DVB-C: <http://en.wikipedia.org/wiki/DVB-C>
- DVB-CI: <http://en.wikipedia.org/wiki/DVB-CI>
- CAM: [http://en.wikipedia.org/wiki/Conditional Access Module](http://en.wikipedia.org/wiki/Conditional_Access_Module)
- King of Sat: <http://en.kingofsat.net/>
- PowerCAM Pro: <http://www.powercam.tv/>
- Aston Professional: <http://www.aston-france.com/uk-modules-professionnels.html>

References

- ✦ V4L-DVB: <http://www.linuxtv.org/>
- ✦ Linux-media ML archives: <http://www.mail-archive.com/linux-media@vger.kernel.org/>
- ✦ libdvbpsi: <http://www.videolan.org/developers/libdvbpsi.html>
- ✦ MuMuDVB: <http://mumudvb.braice.net/>
- ✦ DVBlaster: <http://www.videolan.org/projects/dvblast.html>
- ✦ VLC: <http://www.videolan.org/>
- ✦ MythTV: <http://www.mythtv.org/>
- ✦ IPTV with MythTV: <http://www.avenard.org/iptv/MythTV.html>