

Universidad Técnica Federico Santa María

Departamento de Electrónica

## Transmisión de Webcam utilizando servidor EITV playout de televisión digital

Nombre: Diego Salazar

Profesor: Agustín González

Enero de 2014

## **Introducción**

La televisión digital impone nuevos desafíos de investigación, estos abarcan desde las aplicaciones que se pueden desarrollar con GINGA-NCL, que tienen por finalidad proveer a los usuarios más y mejores servicios, hasta las aplicaciones que pudiesen tener diversos usos y que en el desarrollo de estas permitan además, entender el funcionamiento de la nueva tecnología que será implementada en Chile.

Los nuevos desarrollos deben hacer hincapié en la documentación existente sobre los equipos utilizados, además de estudiar lo que se es capaz de hacer con los protocolos existentes.

## **Resumen**

El presente trabajo pretende documentar el trabajo realizado en los equipos que donó el gobierno de Brasil a la Universidad Técnica Federico Santa María, en el presente informe se presentan las técnicas utilizadas para poder llevar la transmisión de un webcam conectada por USB a un computador, a través de los equipos de los equipos de TV. En el paso hacia la transmisión es necesario indicar que ajustes han sido necesarios para poder llevar cabo esta técnica como así describir las herramientas de software y hardware que fueron utilizadas para el desarrollo de la tarea.

Se detalla también la arquitectura del sistema implementado, de manera de lograr una abstracción de lo que se está realizando, con este diagrama se pueden así obtener las interacciones existentes entre los distintos componentes del sistema

Los resultados obtenidos son finalmente la transmisión de lo captado por la webcam, lo cual queda documentado a través de un video demostrativo, sin embargo, también se precisan que puntos pueden ser mejorados para mejorar la el desempeño de lo que está siendo transmitido y contrastaciones teóricas que pueden inferirse con respecto al uso de otros protocolos.

Los desarrollo que la presente aplicación podría tener son muchísimos, en especial si se lleva a cabo el estudio de lo desarrollado, a webcams ip o a webcams inalámbricas, monitoreo de zonas que quieren resguardo, o simplemente reuniones a distancia son algunas de las aplicaciones que pueden ser desarrolladas, instituciones con infraestructura de gran tamaño podrían aprovechar el desarrollo de esta tecnología, de los modos anteriormente expuestos, si a esto se suma la posibilidad de interactuar con aplicaciones desarrolladas en GINGA-NCL, se puede establecer entonces, un gran abanico de posibilidades para el desarrollo

## **Herramientas utilizadas durante el desarrollo de la transmisión**

A continuación se describen las herramientas utilizadas durante el desarrollo de la experiencia. Se describen fundamentalmente las características más útiles para desarrollar la solución.

Ffmpeg: es una herramienta de software capaz de grabar, convertir un flujo de audio y video, utiliza muchas librerías entre las principales libavcodec, la cual fundamentalmente realiza la codificación de audio y video y la conversión de diversos formatos multimedia al formato transport stream (ts).

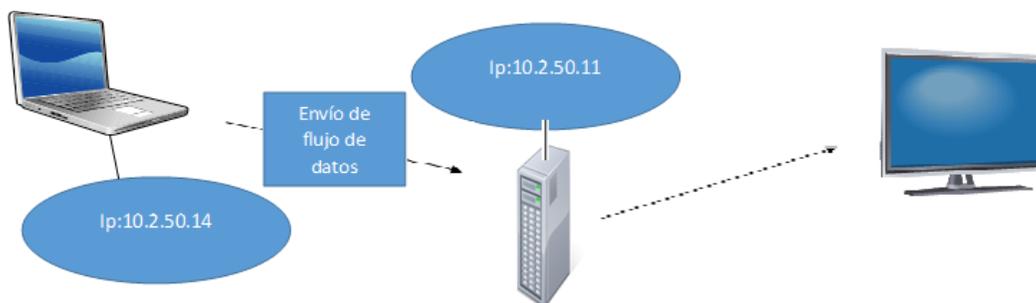
VLAN: reproductor multimedia que posee la capacidad de reproducir streamings de video que se encuentran disponibles entre equipos que forman parte de un mismo segmento de red.

Servidor de televisión digital EITV de playout: servidor que permite el almacenamiento de diversos archivos que será reproducidos mediante la configuración de determinados parámetros se lleva a cabo la reproducción de los archivos.

Televisor Sony: aparato receptor compatible con los estándares desarrollados por la norma brasileña ISDB-Tv.

## **Arquitectura de red y de la solución utilizada utilizada**

A continuación se presenta un diagrama de la arquitectura bajo la cual se llevó a cabo el desarrollo de la presentación



*Ilustración 1 Esquema representativo del sistema*

La arquitectura de red consta de un computador con sistema operativo Ubuntu 12.10 que contaba con los softwares VLC y ffmpeg instalados, y cuya dirección ip es 10.2.50.14. Luego se dispone el servidor playout de televisión digital con dirección ip 10.2.50.11, como se observa tanto el computador como el servidor forman parte de la misma red. Luego se encuentra conectado el televisor que despliega por pantalla la reproducción de los archivos.

En la arquitectura previamente indicada es necesario indicar el comportamiento que tiene cada uno de los componentes del sistema con el objetivo de comprender de una manera más completa la función de cada uno.

Fundamentalmente corresponde a una arquitectura cliente-servidor donde el cliente corresponde al computador que en este caso inicia la comunicación y envía la solicitud que es el flujo de datos, luego el servidor corresponde al servidor EITV playout de televisión digital cuyo objetivo es atender la solicitud que el cliente demanda.

### **Comando ffmpeg y pruebas previas realizadas**

A continuación se presenta el comando que posibilita el envío del flujo de datos desde el pc hasta el servidor

```
ffmpeg -f video4linux2 -s 320x240 -r 25 -i /dev/video0 -f mpegts -muxrate 700k -pix_fmt yuv420p udp://10.2.50.11:8090
```

La expresión `-f video4linux2` establece que el formato de captura de videos sea video4linux2 que básicamente es soportado por cámaras web entre otros dispositivos

La expresión `-s 320x240` hace referencia a la resolución en la cual el video quedará disponible, mayores resoluciones dependen de cada dispositivo.

La expresión `-i /dev/video0` hace referencia a la entrada que se proporcionará, desde la cual se enviará el flujo de datos, en este caso dev/video0, en los sistemas operativos derivados de Linux referencia a dispositivos conectados como por ejemplo una cámara web.

La expresión `-f mpegts` hacer referencia al estándar de transmisión para audio y video almacenado, el cual debe ser convertido a transport stream para poder ser reproducido en el servidor

La sentencia `-muxrate 700k` especifica una tasa de salida de 700kB, es importante fijar que esta tasa sea constante y no variable. Se pueden permitir fluctuaciones de unos pocos kB pero nunca del orden de las decenas.

La expresión `-pix_fmt yuv420p` indica que se utilizará el modelo matemático YUV para la representación de los colores

Finalmente se indica la dirección ip de la red en la cual el video quedará disponible y el protocolo que será utilizado en este caso udp.

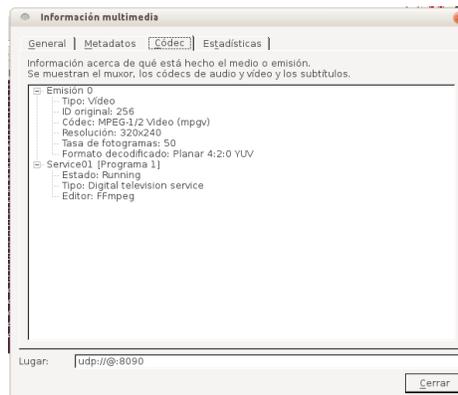
### **Utilización de VLC**

Una vez ejecutado el comando es recomendable utilizar el reproductor VLC para poder extraer ciertos parámetros que son necesarios en el servidor playout, para esto se puede ejecutar el comando especificando la dirección ip del computador en el cual se está trabajando.

Luego se selecciona en vlc la opción abrir volcado de rede y se escribe la dirección abreviada que indica que el flujo es el propio computador de trabajo `udp://@8090`



*Ilustración 2 Salida del volcado de red visto por ffmpeg*



*Ilustración 3 Datos obtenidos utilizando VLC*

### **Configuración de servidor EITV de playout**

Una vez obtenidos los datos mediante vlc se procede a realizar las configuraciones necesarias para que el flujo de datos sea reproducido por el servidor.

Multiplexer	
<b>AIT multiplexing</b> Enable/Disable Application Information Table (AIT) multiplexing (default: On)	<input checked="" type="checkbox"/>
<b>BIT multiplexing</b> Enable/Disable Broadcaster Information Table (BIT) multiplexing (default: On)	<input checked="" type="checkbox"/>
<b>CAT multiplexing</b> Enable/Disable Conditional Access Table (CAT) multiplexing (default: On)	<input checked="" type="checkbox"/>
<b>EIT multiplexing</b> Enable/Disable Event Information Table (EIT) multiplexing (default: On)	<input checked="" type="checkbox"/>
<b>NIT multiplexing</b> Enable/Disable Network Information Table (NIT) multiplexing (default: On)	<input checked="" type="checkbox"/>
<b>PAT multiplexing</b> Enable/Disable Program Association Table (PAT) multiplexing (default: On)	<input checked="" type="checkbox"/>
<b>PMT multiplexing</b> Enable/Disable Program Map Table (PMT) multiplexing (default: On)	<input checked="" type="checkbox"/>
<b>SDT multiplexing</b> Enable/Disable Service Description Table (SDT) multiplexing (default: On)	<input checked="" type="checkbox"/>
<b>SDTT multiplexing</b> Enable/Disable Software Download Trigger Table (SDTT) multiplexing. (default: On)	<input checked="" type="checkbox"/>
<b>TMCC Remultiplexing</b> Enable/Disable TMCC Remultiplexing. When enabled, output will be a 204-byte BTS according to ARIB STD-B31 5.5. Disabling this will cause output to be a 188-byte TS, without the 16 dummy control bytes. (default: On)	<input checked="" type="checkbox"/>
<b>TOT multiplexing</b> Enable/Disable Time Offset Table (TOT) multiplexing (default: On)	<input checked="" type="checkbox"/>

*Ilustración 4 Configuraciones de multiplexación hechas al servidor*

Todas las opciones de multiplexación deben estar habilitadas por defecto

Luego se procede llenar los campos con los parámetros solicitados

Para ello se debe ingresar a la sección Audio/ Video de la interfaz del playout, e indicar cual será la fuente de nuestro flujo junto al protocolo y la dirección ip.

Además se debe indicar el identificador del video y un identificador de audio, además de las codificaciones utilizadas para ambos, para el caso no se utilizó ninguna fuente que capturara audio por lo cual, se procedió a asignar un identificador válido.

También se debe señalar el Program Clock Reference que típicamente referencia al video, también es necesario señalar la tasa en bits que en el comando de ffmpeg indicaba que era constante, para el caso 700000 bits

Stream ID **87**

Name

Source  IP

Address:

IP Interface  Port:

Protocol:  UDP  RTP

PID Video

Video Stream Type

PID Audio

Audio Stream Type

PID PCR

TOT time decode  Synchronize system time with TOT from this input

Bitrate Video (kbps)

Bitrate Audio (kbps)

Bitrate TS (bps)

*Ilustración 5 Configuración de audio/video hecha al servidor*

Luego es necesario asignar a un servicio el video que anteriormente se ha parametrizado

Name

Provider

Service Type

EIT Schedule Flag

EIT Present Following Flag

H-EIT Flag

M-EIT Flag

L-EIT Flag

**Custom**

The following parameters only need to be configured if custom values are needed. The value "0" (zero) can be left for defaults.

**Program PIDs**

Primary Video PID  (decimal)

Video Descriptors

Primary Audio PID  (decimal)

Audio Descriptors

PCR PID  (decimal)

**AIT**

Automatic Schedule Operation  When Application Scheduling is ON, this option automates the life-cycle of Applications (control codes are sent automatically).

PID  (decimal)

Repetition Rate  ms

**DSM-CC Object Carousel**

PID  (decimal)

*Ilustración 6 Configuración de servicios hecha al servidor*

Name	Action
Video Camara	 

**Associate Stream**

No stream matching AV PIDs and bitrates has been found.

ack

Ilustración 7 Asociación de audio/video a un servicio

Luego se debe realizar la configuración de la transmisión del video, de los parámetros previamente seleccionados lo importante es indicar el tipo de modulación que será utilizada, como se requiere que la reproducción sea de la mayor calidad posible a través de un receptor estático se utilizó 64-QAM, además se aprecia que asignan a esta capa 12 segmentos, en el caso desarrollado se observa un segmento que se dejó disponible para la transmisión de un dispositivo móvil

 **Transmission Configuration**

Transport Stream: webcam\_usb

Configuration
DVB-C
DVB-T
DVB-S
ISDB-T

---

**Broadcast type** 13-segment TV Broadcast

**Frequency** Channel 25 539142857 Hz

**Area Code** 1

**Virtual Channel (Remote Control Key ID)** 1

**TS Name** E!TV Playback

**Transmission Mode** Mode 3: 8k

**Guard Interval** 1/16

**Partial Reception**  Enabled partial reception in the Layer A (implies in Layer A with only 1 segment)

**Emergency**  Enabled emergency flag

**Hierarchical Layers**

	Layer A	Layer B	Layer C
<b>Segments</b>	1	12	0
<b>Modulation</b>	QPSK	64-QAM	DQPSK
<b>Convolutional Rate</b>	2/3	3/4	5/6
<b>Time Interleaving Length</b>	4	2	8
<b>Bitrate (bps)</b>	440560	17842800	0
<b>Total Bitrate (bps)</b>	18283360		

Save

Ilustración 8 Configuración de transport stream hecha al servidor

Luego de eso de eso al seleccionar play en el servidor se puede observar lo realizado y que despliega en pantalla

### **Demostración**

El link de la demostración está disponible en:

<http://www.youtube.com/watch?v=3cyzid4ZF8Y>

### **Conclusiones y trabajo futuro**

Los resultados que entrega la demostración afirman el cumplimiento de los objetivos planeados, es decir se comprobó el funcionamiento de la transmisión de un streaming en tiempo real sin embargo, existen mejoras que es posible llevar a cabo con respecto a la disminución de los retardos, una manera de lograr disminuir dicho retardo es con la utilización de los mismos comandos de ffmpeg.

La utilización del protocolo UDP es necesaria en la transmisión, dado que el hecho de perder unos cuantos paquetes no afecta a la recepción de manera global del contenido del video sin embargo, una opción interesante sería experimentar con el protocolo RTP, y experimentar las diferencias notables que entre ellos existen.

### **Bibliografía**

[www.ffmpeg.org](http://www.ffmpeg.org)

-Manual EITV playout.