



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

**Desarrollo de un sistema de
automatización de máster para
televisión**

INFORME DE ACTIVIDADES PROFESIONALES

Que para obtener el título de
Ingeniero en Computación

P R E S E N T A

Ernesto Espinosa Ponce

ASESOR DE INFORME

M.C. Alejandro Velázquez Mena



Ciudad Universitaria, Cd. Mx., 2019

Contenido

Introducción	1
Capítulo 1. Organigrama del área	3
Capítulo 2. Descripción de proyectos.....	7
Capítulo 3. Desarrollo de un sistema de automatización de máster para televisión	19
3.1 Breve historia de la televisión	19
3.2 Sistemas de automatización de máster para televisión.....	22
3.3 Objetivo del proyecto	28
3.4 Subsistema de ingestión.....	31
3.4.1 Diseño de la base de datos para la ingestión de materiales.....	38
3.4.2 Módulo interfaz WEB de alta de programas	43
3.4.3 Módulo interfaz WEB de alta de capítulos.....	44
3.4.4 Módulo interfaz WEB ingestión de programas	45
3.4.5 Módulo transcodificación de programas.....	48
3.4.6 Almacenamiento de los programas en baja resolución	53
3.4.7 Interfaz WEB para consulta de programas ingestados.....	54
3.4.8 Módulo interfaz WEB ingestión de spots y bloques de programa.....	55
3.4.9 Módulo transcodificación automática de spots	57
3.4.10 Módulo catálogo de spots ingestados	58
3.4.11 Módulo de Ingestión manual	59
3.4.12 Herramientas auxiliares	61
3.5 Subsistema de almacenamiento	63
3.5.1 Base de datos del subsistema de almacenamiento.....	66
3.5.2 Módulo envío automático de alta resolución de programas.....	68
3.5.3 Módulo envío automático de spots transcodificados	69
3.5.4 Módulo de ingestión manual	71
3.5.5 Herramientas de depuración del subsistema de almacenamiento.....	71
3.6 Módulo de grabación	74
3.6.1 Herramienta de depuración automática del módulo de grabación.....	84
3.7 Módulo de extracción de grabaciones.....	85
3.8 Módulo de reproducción	87
3.8.1 Servicio de interfaz con el hardware	87

3.8.2 Reproductor de Máster	120
3.8.3 Depurador del reproductor de máster	161
Capítulo 4. Resultados.....	163
Conclusiones	167
Conclusiones personales.....	168
Glosario	171
Referencias	181

Índice de figuras

Figura 1 Organigrama del área	5
Figura 2 Módulos propuestos para sistema de automatización de máster	30
Figura 3 Niveles de información para operación de un canal de televisión	32
Figura 4 Estructura de una guía de programación	33
Figura 5 Módulos del subsistema de ingestión para ingreso de programas completos	36
Figura 6 Módulos del subsistema de ingestión para spots y bloques individuales.....	37
Figura 7 Módulos del subsistema de almacenamiento.....	64
Figura 8 Diagrama de conexiones de la tarjeta BlackMagic Decklink Quad	75
Figura 9 Direccionamiento de espacio de color UYVY en memoria	77
Figura 10 Diagrama de bloques de módulo de grabación de señales diferidas.....	81
Figura 11 Relación de imágenes I,B y P en el estándar MPEG	83
Figura 12 Diagrama de funcionamiento del módulo de reproducción.....	87
Figura 13 Arquitectura del servicio de interfaz con el hardware	89
Figura 14 Arquitectura del reproductor de máster	122

Índice de Tablas

Tabla 1 Diseño de la base de datos para entrega de materiales.....	38
Tabla 2 Estructura tabla base de datos TVP_CANALES	39
Tabla 3 Estructura tabla base de datos TVP_PROGRAMAS	39
Tabla 4 Estructura tabla base de datos TVP_CAPITULOS	39
Tabla 5 Estructura tabla base de datos TVP_BLOQUES_PROG	41
Tabla 6 Estructura tabla base de datos TVP_CODIFICADORES	42
Tabla 7 Estructura tabla base de datos TVP_PETICION_REINGES_PROG	42
Tabla 8 Estructura tabla base de datos TVP_EDO_PET_REINGES	42
Tabla 9 Estructura tabla base de datos TVP_ALMAC_WEB	43
Tabla 10 Estructura tabla base de datos TVP_MAT_TRANS	66
Tabla 11 Estructura tabla base de datos TVP_ESPACIOS_ALMACENAMIENTO	67
Tabla 12 Almacenamiento de muestras de audio en memoria	79

Introducción

La televisión es uno de los medios de comunicación con mayor cantidad de usuarios alrededor de todo el mundo. En México de acuerdo a información del Instituto Federal de Telecomunicaciones, el 96% de la población tiene acceso a la señal de televisión abierta y el 66% de los hogares cuenta con servicios de televisión de paga. Esto se traduce en millones de espectadores a los cuales puede llegar la señal de televisión emitida por alguna cadena.

La industria de la televisión genera ventas en nuestro país del orden de miles de millones de pesos anualmente y las cadenas de televisión tienen márgenes de ganancia por lo menos de cientos de millones de pesos, es por lo tanto una industria que genera fuentes de empleo y de ingresos aportando tanto a la economía nacional como a los trabajadores que en ella se emplean.

Como cualquier otra industria las empresas pertenecientes a este ramo se encuentran en constante competencia. Por otro lado, con la llegada de nuevos medios de comunicación y entretenimiento que utilizan Internet, tales como los sistemas de video bajo demanda, las redes sociales, plataformas de distribución de video como YouTube y todo el contenido en general al cual se puede acceder no solo a través de una computadora en casa sino también a través de un dispositivo móvil como un Smartphone, una tablet e incluso desde las SmartTV o dispositivos similares, dichas empresas se enfrentan hoy en día a nuevos desafíos para cautivar a las audiencias compitiendo con estos nuevos medios de comunicación. Los retos de una empresa dedicada a la transmisión de televisión se extienden desde la producción de los contenidos hasta la forma en la cual se distribuyen los mismos para hacerlos llegar a los espectadores, de forma que sin la adopción de nuevas tecnologías una empresa quedaría rezagada, superada por la competencia y sin posibilidad de subsistir como en cualquier otra actividad comercial.

La televisión desde sus orígenes a principios del siglo XX se ha apoyado siempre en los avances tecnológicos. Originalmente se utilizaron métodos mecánicos para adquirir y reproducir imágenes, posteriormente se adoptaron sistemas electromecánicos continuando con el avance tecnológico hacia el uso de dispositivos electrónicos. El desarrollo de la computación trajo consigo medios de almacenamiento con mayor capacidad y velocidad de respuesta, equipos con mayor capacidad de procesamiento, redes de datos y sistemas digitales que se pueden aprovechar en gran cantidad de procesos. La televisión, principalmente en la década de los 90, encontró un poderoso aliado en las herramientas de cómputo para hacer evolucionar gran parte de sus procesos incluyendo desde la creación de contenido hasta las operaciones necesarias para que la señal llegue hasta el televidente en la comodidad de su casa. Hoy en día gran parte del quehacer cotidiano en la operación de una televisora es inconcebible sin los sistemas de cómputo especializados para todas las áreas de operación.

Muestra de esto es el llamado “apagón analógico” que se llevó a cabo nuestro país en el año 2015 y consistió en la adopción de la tecnología de la Televisión Digital Terrestre (TDT) para la transmisión de la señal de televisión abierta; esto requirió inversiones de tiempo, dinero e investigación por parte de las empresas de televisión para llevar a cabo esta transformación. Los equipos encargados de procesar la señal para su transmisión en las antenas repetidoras a lo ancho de la república están basados en sistemas computacionales de procesamiento de señales y redes de datos. Las tecnologías de compresión de señales de audio y video son de vital importancia hoy en día en todos los procesos de una televisora así como los avances en la distribución de contenidos apoyada en redes de datos.

La automatización de procesos también es un área de oportunidad que debe ser explorada en toda empresa con la finalidad de reducir costos y aumentar los márgenes de ganancias, también ayuda a optimizar los procesos reduciendo los errores debidos a la operación humana. Los sistemas de cómputo son herramientas básicas para lograr este objetivo en cualquier empresa.

Por todo lo anterior es que dentro del ramo de la televisión se requiere de ingenieros en computación calificados y profesionales para responder a todas las demandas y retos a los que se están enfrentando actualmente las televisoras. El contar con un profesional en el campo de la computación que sea capaz de investigar nuevas tecnologías, evaluar las ventajas y desventajas de la adopción de las mismas, realizar el desarrollo de sistemas a la medida de las necesidades para empresas de televisión que generalmente son altamente especializadas, diseñar software y soluciones de la más alta calidad, organizar equipos de trabajo y tomar decisiones estratégicas, entre algunas otras tareas, es muy importante para mantener y mejorar las operaciones más importantes.

En el presente informe se expone el desarrollo de un sistema de automatización de máster para la transmisión de canales de televisión. Este tipo de sistemas existen de manera comercial, sin embargo, debido a la cantidad de cadenas televisivas que existen son altamente especializados y la inversión para adquirirlos es muy alta, ni hablar de las inversiones requeridas para realizar una adecuación a la medida. Un sistema de automatización de máster permite emitir una señal de televisión de forma ininterrumpida con la menor cantidad de personal posible para operar. Un sistema de automatización a muy grandes rasgos cuenta con componentes para ingresar los contenidos audiovisuales al sistema, almacenarlos y mantenerlos catalogados así como para reproducirlos en el momento adecuado según una lista de reproducción definida para un canal de televisión.

El desarrollo de estos sistemas hechos en México reduce considerablemente la inversión para su adquisición y adecuaciones; por otro lado maximiza las ganancias del negocio y optimiza los procesos para reducir el margen de error. La creación de este sistema también enfrentó otros retos como la interacción con equipos de desarrollo de otros sistemas, realizar un buen diseño de forma que sea fácil de mantener y realizar adecuaciones, plantear ideas que mejoren la operación de los usuarios y muchos otros.

Capítulo 1. Organigrama del área

La empresa en la cual desempeñé las actividades relacionadas con el proyecto del presente informe desarrolla soluciones a la medida para empresas de la industria de la televisión en México. Las soluciones desarrolladas forman parte de los avances tecnológicos disponibles para clientes en materia de transmisión de televisión y distribución de la señal generada.

La industria de la televisión es un mercado muy competido que se transforma constantemente con el paso del tiempo, la adopción de los avances tecnológicos en las operaciones representa incrementos en la competitividad y productividad gracias a las ventajas que representan los mismos.

En los inicios de la televisión en México la tecnología disponible era análoga hasta avanzada la década de 1990, cuando empezaron a aparecer los primeros medios digitales de transmisión así como importantes avances dentro de la computación que a la larga permitirían el aprovechamiento de equipos de cómputo, redes de datos, dispositivos de almacenamiento, software de edición, hardware de captura y procesamiento de señales así como la aplicación de muchas otras tecnologías sin las cuales es prácticamente inconcebible el trabajo que se realiza en una televisora en la actualidad.

La empresa en la cual desarrollé este proyecto cuenta con un Departamento de Desarrollo Tecnológico, el cual se encuentra dentro de la Dirección de Ingeniería. Esta dirección es la encargada de la adopción de nuevas tecnologías, el desarrollo de tecnologías propias, las mejoras en los procesos, operación y el mantenimiento de la infraestructura existente.

En el Departamento de Desarrollo Tecnológico he desarrollado proyectos desde hace 8 años pertenecientes a la rama de la ingeniería. El Departamento de Desarrollo Tecnológico desarrolla software y sistemas de cómputo a la medida dedicados a las tareas de operación trascendentales para empresas que realizan transmisión de televisión. En este departamento se han creado sistemas que son de uso general para la transmisión de televisión en todo el territorio nacional así como en otros países. Algunos de los proyectos creados han dado lugar a ideas que continúan siendo desarrolladas por otras áreas.

En el área de Departamento de Desarrollo Tecnológico desarrollamos tecnología mexicana que compite con cualquier solución disponible en el mercado, con la ventaja de reducir los costos, ya que adquirir tecnología para la industria de la televisión generalmente implica grandes inversiones monetarias para la adquisición de equipo dedicado, inversión para el escalamiento de características, así como planes de soporte con un costo adicional cada cierto período de tiempo.

El Departamento de Desarrollo Tecnológico se divide en dos secciones: Soporte Técnico y Desarrollo.

La sección de Soporte Técnico se encarga de ayudar a los usuarios finales de nuestros sistemas a resolver las dudas y problemas que se presenten en las soluciones desarrolladas e implementadas que generan inconvenientes en la operación diaria debido a defectos en los sistemas, problemas de comunicación entre equipos desplegados en todo el territorio nacional, problemas en el hardware o cualquier otra problemática. Otras funciones de esta sección son el mantenimiento de los equipos de cómputo y el hardware que utilicen las aplicaciones, la instalación y configuración del software en los equipos y el monitoreo de los equipos en todo el territorio nacional para asegurarse que funcionen de forma autónoma sin supervisión en sitio.

En la sección de Desarrollo se encuentran todos los integrantes del equipo de trabajo que diseñan, desarrollan, programan, dan mantenimiento e implementan soluciones para la empresa. El desarrollo de estas soluciones generalmente involucra el análisis de requerimientos, diseño de sistemas, programación y mantenimiento de software cuya naturaleza es muy variada, la cual se expondrá más adelante. Gran parte de las aplicaciones requieren integrar hardware a bajo nivel, por lo que en su mayoría se programan en lenguaje de programación C/C++, dado que muchos fabricantes cuentan con SDK para estos lenguajes. Las aplicaciones también requieren realizar conexiones a bases de datos para guardar y recuperar información. Otro gran grupo de aplicaciones realiza el procesamiento de señales de audio y video, compresión de datos, y la transmisión de esas señales tanto en banda base como a través de redes de datos IP. También se desarrollan aplicaciones para la distribución de información sobre redes de datos ya sean Ethernet o vía satélite.

La sección de desarrollo cuenta con un laboratorio donde se realizan las investigaciones necesarias tanto con el hardware como las tareas de desarrollo de software. El trabajo diario incluye además de las tareas inherentes al desarrollo de software la búsqueda de información de estándares internacionales principalmente en materia de transmisión de televisión, investigaciones acerca de nuevo software así como la actualización de metodologías y técnicas de programación.

En general el trabajo dentro del Departamento de Desarrollo Tecnológico es muy variado e interesante ya que siempre se están ideando mejoras en los proyectos existentes y proyectos nuevos que ayuden a incrementar la productividad de las empresas televisivas y por lo tanto su competitividad tanto en el entorno nacional como internacional.

El organigrama del Departamento de Desarrollo Tecnológico es el siguiente:

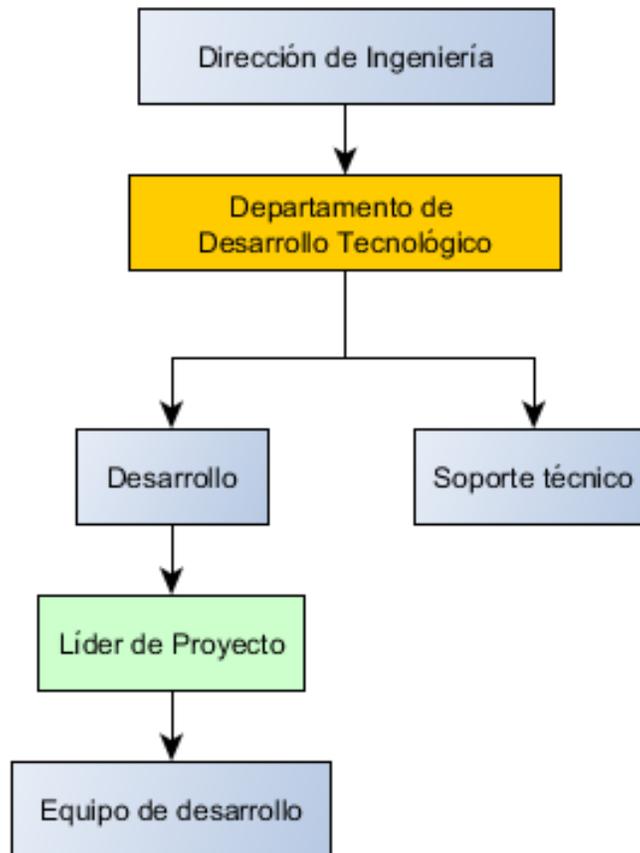


Figura 1 Organigrama del área

Inicialmente ingresé en el puesto Desarrollador de Software, posteriormente fui ascendido a Líder de Proyecto con un equipo de 5 desarrolladores a mi cargo. Al momento de presentar este informe también soy el supervisor de la sección de soporte conformada por 10 elementos.

El trabajo diario del puesto de Líder de Proyecto implica una serie de tareas entre las que se encuentran:

- Organización del equipo de trabajo
- Coordinación y seguimiento de proyectos
- Verificar el cumplimiento de las metas del equipo en los plazos establecidos
- Levantamiento y análisis de requerimientos para nuevos proyectos.
- Definición del alcance de los proyectos.
- Diseño de los nuevos sistemas y aplicaciones

- Interacción directa y reporte de resultados con los clientes que solicitan los proyectos de desarrollo.
- Desarrollo y programación de módulos, aplicaciones y sistemas
- Análisis, investigación y adopción de nuevas tecnologías
- Implementación y despliegue de las soluciones ya terminadas
- Generar nuevas propuestas de proyectos para cubrir necesidades de la empresa
- Identificar y controlar los riesgos del diseño de los sistemas
- Diseño de procesos de desarrollo y soporte
- Supervisión y ejecución de pruebas de los sistemas desarrollados
- Reclutamiento y capacitación de nuevo personal
- Generación de planes de mantenimiento y supervisión de la ejecución.
- Reporte de resultados semanales a la gerencia y Dirección de Ingeniería.

Capítulo 2. Descripción de proyectos

Durante mi trayectoria en el Departamento de Desarrollo Tecnológico he colaborado en la realización, desarrollo e implementación de varios proyectos que se describen a continuación.

Herramienta de Análisis de Contenidos

La Herramienta de Análisis de Contenidos fue el primer proyecto en el cual participé. Fue un sistema WEB desarrollado con Java Enterprise Edition utilizando Java Beans y conexiones para consultas a una base de datos Oracle en Hibernate. La Herramienta permitía mediante su interfaz web capturar una calificación analítica de los capítulos de un programa de televisión con la finalidad de estudiar el impacto que tenían en el público televidente una vez que eran transmitidos. Con base en la calificación el sistema era capaz de generar una serie de reportes con diferentes vistas, gráficas y conteos. En este proyecto participé realizando mantenimientos a la sección de los reportes programando en lenguaje Java bajo el entorno de desarrollo Eclipse para añadir nuevas secciones a los reportes, modificaciones en la interfaz gráfica y arreglar errores dentro del sistema. Actualmente se encuentra discontinuado, fue remplazado por otro sistema diseñado por mi equipo de trabajo.

Sistema de Almacenamiento de Acervo de Video

El Sistema de Almacenamiento de Acervo de Video fue un sistema dedicado a la adquisición de video desde medios de almacenamiento en cinta, conversión del material a archivos digitales para su almacenamiento, calificación del material, envío del material a un sistema de almacenamiento masivo y un visor para la búsqueda así como descarga de material en formato digital para poder ser utilizados dentro de la edición de programas de televisión. En este proyecto realicé el mantenimiento principalmente de la aplicación de búsqueda, la cual permitía buscar los videos almacenados en formato digital, visualizarlos en el equipo del usuario y también era posible solicitar la descarga de los mismos para poder utilizarlos en las diferentes producciones. Las actividades de mantenimiento incluyeron corregir errores de programación en la aplicación, desarrollé un módulo de texto predictivo para agilizar las búsquedas, se optimizaron las búsquedas en la base de datos para reducir el tiempo de respuesta y se mejoró la seguridad del sistema basado en las recomendaciones de un análisis realizado por un área dedicada a la seguridad de la información y aplicaciones.

Realicé también mantenimiento al módulo de descargas y el módulo de envío al sistema de almacenamiento masivo. El sistema de almacenamiento masivo es un sistema propietario el cual cuenta con un API de programación basada en una arquitectura cliente - servidor la cual se utiliza para enviar los materiales al almacenamiento final. El API recibe actualizaciones por parte del fabricante regularmente cada 3 o 4 años, por lo que es necesario actualizar las aplicaciones de almacenamiento en ese mismo periodo.

El desarrollo está hecho en lenguaje C/C++ además del uso del API de Quicktime a bajo nivel así como la integración de conexiones a una base de datos Oracle.

Actualmente este sistema ya no se encuentra en uso, cumplió su ciclo de vida y fue reemplazado por un sistema que incluye nuevas tecnologías para la adquisición y distribución de video.

Sistema de almacenamiento de Audio Digital e Informes de Uso

El Sistema de almacenamiento de Audio Digital e Informes de Uso es un conjunto de programas de cómputo diseñados para almacenar y administrar el acervo digital de audio con el cual cuenta una televisora así como el registro y reporte del uso de la música dentro de los programas televisivos que son transmitidos. El sistema fue programado en lenguaje C/C++ con integración de conexiones a un servidor de base de datos Oracle así como el uso del protocolo de transferencia de archivos FTP, integración de envío de correos tanto a cuentas de correo electrónico empresariales como correos externos mediante la integración del protocolo SMTP. Cuenta con un sistema de ingestión de media distribuido, un buscador de material para clientes finales con reproducción de previo integrada y también permite solicitar la descarga de audio en formatos digitales de alta calidad.

En este sistema las actividades en las que participé fueron:

- Mantenimiento de las aplicaciones del sistema.
- Mejora de la seguridad en las aplicaciones, incluyendo la aplicación de algoritmos de encriptación en transacciones de acceso al sistema, transacciones críticas, en las bitácoras del sistema (logs) con los registros de acciones de usuarios dentro del sistema y archivos de configuración.
- Mejora a los flujos de trabajo y procesos de las aplicaciones
- Diseño, desarrollo e implementación de nuevas características para satisfacer las necesidades de las áreas que operan el sistema. Estas modificaciones van desde nuevos tipos de informes y reportes así como la creación de nuevos metadatos acerca de los temas musicales.
- Mejora a los procesos de soporte
- Interfaz con los servidores de correo empresariales en protocolo SMTP.

Página WEB del Área de Adquisiciones

La página Web del Área de Adquisiciones es un plataforma en web programada en ASP con C# basado en el framework .NET de Microsoft. Fue diseñada para facilitar el trabajo del área de adquisiciones para empresas de televisión gestionando toda la información que maneja esta área en un solo portal Web. Cuenta con un catálogo completo de programas de diferentes productoras de contenido en todo el mundo, las cuales se estudian en las áreas de adquisiciones y se lleva a cabo el registro de los mejores candidatos en la página, lo anterior para dar seguimiento a la información necesaria para adquirirlos y transmitirlos. La página tiene una interfaz de administración completa de

catálogos de los programas y permite guardar datos relativos a los mismos como lo es países de origen, productoras, cadenas televisivas de origen, entre otros. También permite adjuntar información publicitaria de los programas tales como trailers, pósters originales de las producciones, imágenes oficiales, documentos con análisis de rating, informes etc. Las actividades que realicé en esta página fueron las siguientes:

- Mantenimiento de la página arreglando los bugs reportados por los usuarios.
- Mejora del monitoreo de operaciones que realizan los administradores de la página para conocer exactamente quién realiza los cambios y cuáles fueron.
- Monitoreo de las secciones y programas más solicitados por los visitantes del sitio.
- Soporte para nuevos formatos de video para los trailers así como creación de nuevas secciones para registro los estrenos de producciones en otras partes del mundo.
- Migración del framework .NET con el que originalmente fue programada en el 2003 a versiones más nuevas de Microsoft Visual Studio, con lo que se ha logrado mantener la página vigente al día de hoy.

Página WEB Noticias Breves

La página WEB de Noticias Breves es un sitio web para la publicación de noticias relevantes de la industria de la televisión, cinematográfica y la escena musical principalmente. Está diseñada en forma de un periódico digital interno para empresas televisoras. En este proyecto realicé el análisis de requerimientos, diseño y supervisión del desarrollo.

A lo largo del tiempo de vida del proyecto han existido 4 versiones diferentes con cambios drásticos en el diseño visual así como funcionalidades dependiendo de la empresa cliente. Además han participado 5 programadores en este proyecto a los cuales he supervisado y coordinado el desarrollo que llevan a cabo. Este proyecto fue desarrollado en lenguaje PHP y JavaScript, puede ser instalado en un servidor Apache y utilizar la base de datos MySQL.

El proyecto continúa constante mejora, ha estado a mi cargo desde hace 6 años.

Sus principales características son:

- Es un periódico semanal con noticias y otros datos relevantes.
- Tiene varias secciones para publicar noticias acerca de televisión, música, cine y otras notas breves. Cada noticia puede contener una imagen ilustrativa.
- Cuenta con soporte para publicar un video semanal.
- En una sección se pueden publicar tablas del comportamiento en taquilla de producciones cinematográficas en México y Estados Unidos, cada registro puede incluir su video de avances.
- Sección de comentarios disponible para los visitantes del sitio.
- Búsqueda de noticias por palabras clave en todo el histórico de semanas desde que se creó el sitio web.

- Interfaz de administración para registrar las noticias semanalmente, cartas de programación de otros países con rating, cartelera de estrenos cinematográficos mensuales en México, tablas de comportamiento en taquilla de producciones cinematográficas tanto en México como Estados Unidos

La interfaz de administración además permite revisar una vista previa del sitio para el número de semana que uno desee y así ajustar el contenido que se va a publicar. Una vez que el administrador está satisfecho con la vista aprueba el contenido para que esté disponible instantáneamente a los visitantes del sitio.

Sistema de Análisis de Guiones

El Sistema de Análisis de Guiones fue un proyecto desarrollado para prever el éxito que tendría un programa televisivo en la audiencia. En este proyecto se trabajó en colaboración con un equipo de psicólogos y sociólogos los cuales crearon una metodología para analizar el contenido de los guiones basado en las acciones que realizaban los personajes e interacciones entre los mismos, de esa manera se podría prever desde la lectura del texto de un guión si sería interesante o no para cierto grupo de audiencia.

El objetivo de este proyecto es sumamente importante debido a que no es necesario llevar a cabo los gastos que representa la grabación de un programa así como su edición y postproducción para luego transmitirlo y entonces revisar la reacción de la audiencia, dependiendo de esta reacción es que se toman decisiones estratégicas en el contenido de las producciones televisivas para generar un mayor gusto en el público o incluso puede ser cancelado. Este sistema evita todo este ciclo así como la inversión necesaria para llegar al punto de la toma de decisiones.

Con este sistema y metodología se realizaron algunos de los proyectos más exitosos de en televisoras. Además es el sucesor de la Herramienta de Análisis de Contenidos dadas las ventajas antes descritas.

El sistema consiste en una página Web desarrollada con lenguaje Java, en particular con la tecnología JSP, JavaScript y aplicación de CSS3 el cual se ejecutaba en un servidor JBoss sobre un sistema operativo Windows. La base de datos utilizada fue Oracle. El sistema permite la carga de guiones de producciones televisivas organizados por capítulos, esos guiones son entregados en formato de Microsoft Word y el sistema los convierte a HTML. Una vez almacenados en este formato se despliegan mediante una página web al usuario para su lectura y análisis. La página de despliegue además de visualizar el guión permite capturar las acciones que va realizando cada personaje así como clasificarlas según un catálogo propuesto por la metodología.

Una vez que se tienen estos elementos en la base de datos el sistema puede realizar diferentes conteos y reportes, cruces entre acciones de personajes y generar resúmenes semanales, mensuales o de todos los guiones de la producción. También cuenta con un módulo que permite obtener gráficas de la información en diferentes vistas.

Los resultados del sistema se utilizan en el análisis del impacto que tendría en la audiencia con aspectos tales como el ritmo de la historia o una medida cuantitativa de lo emocionante que podría ser para el televidente. La metodología incluye valores de referencia que son incluidos a manera de benchmark en los reportes. Finalmente permite guardar el video del capítulo tal como fue grabado para realizar comparaciones contra el guión original, ya que puede haber diferencias introducidas al momento de la grabación o en la edición del capítulo.

Sistema automático para inserción de comercialización

Una de las principales actividades generadoras de ingresos en una televisora es la transmisión de comercialización. Existe en la industria de la televisión el concepto de inserción de comercialización regional, esto es la transmisión de comercialización que puede ser diferente en las distintas ciudades y poblaciones del país e incluso en otros países donde se tenga cobertura.

En los inicios de la televisión a nivel mundial, se contaba en los sitios donde estaba instalada la antena transmisora para cierta población con empleados que bajo un itinerario iniciaban la reproducción de los materiales regionales. Con el avance de la tecnología esto fue cambiando en todo el mundo, lo cual dio paso a la creación de sistemas de inserción de comercialización automática muy sofisticados y de reconocidas marcas líderes en el ramo de la industria de la televisión. Sin embargo este tipo de sistemas tienen un alto costo tanto para la adquisición, actualización y el soporte que requieren, por otro lado si se necesita el desarrollo de nuevas características también representa una fuerte inversión y tiempo de espera ya que en su mayoría las marcas líderes tienen sus oficinas centrales y de desarrollo en otros países. Por ello se ha desarrollado a lo largo de los años un sistema capaz de insertar los comerciales regionales dentro de una señal según las necesidades de televisoras mexicanas y que lleve a cabo automáticamente las tareas que van desde la distribución del material, inserción de comercialización, monitoreo, autodiagnóstico y entre otras más.

A lo largo de los años he trabajado en todos los módulos que integran este sistema realizando el análisis de nuevos requerimientos, diseño, desarrollo, programación, implementación y mantenimiento de aplicaciones tanto ya existentes como nuevas. El sistema se encuentra desarrollado en lenguaje de programación C/C++ casi en su totalidad.

Algunos de los módulos en los que he participado son los siguientes:

- **Capa de software entre el reproductor y el hardware del equipo de inserción en antena transmisora.** El sistema cuenta con un módulo que realiza la decodificación de los videos almacenados en formato digital y envía los datos decodificados de audio y video a un servicio de Windows encargado de comunicarse con el hardware utilizado para la generación de la señal de televisión en banda base. El servicio de Windows constituye la capa de software entre el reproductor y el hardware, participé en su diseño y programación. La capa de

software de comunicación con el hardware además se ha aprovechado en otros proyectos que requieren la generación de señal de televisión en banda base. El servicio realiza el procesamiento de la señal todo el tiempo sin interrupciones e incluye mecanismos de emergencia para tomar acciones correctivas ante cualquier error, es una aplicación de alto rendimiento y disponibilidad.

- **Módulo de ingestión de materiales.** El sistema cuenta con un módulo que permite grabar los comerciales a transmitir desde una VTR profesional o puede importar un video en un archivo digital. Al importar el módulo revisa que las propiedades del archivo cumplan con un estándar establecido. Bajo ciertas configuraciones es capaz de realizar la recodificación de archivos digitales a un formato que cumpla con el estándar definido para el sistema. El sistema cuenta con una base de datos para almacenar los metadatos de los materiales ingestados. Participé en el diseño, programación y mejoras de este módulo.
- **Reproductores de materiales.** El sistema cuenta con aplicaciones de reproducción para realizar tareas de monitoreo y revisión de los materiales ingestados. Participé en el diseño, programación e implementación de reproductores basados en tecnologías como QuickTime de Apple y DirectShow de Microsoft.
- **Distribución de materiales vía satélite.** Realicé mejoras en el diseño de las aplicaciones para la distribución de materiales de video y envío de instrucciones del sistema vía satélite a estaciones repetidoras remotas.
- **Monitoreo de estaciones remotas.** Desarrollé aplicaciones para el monitoreo de estaciones remotas y la optimización de los procesos de comunicación entre servidores centrales de monitoreo y equipos en antenas transmisoras.
- **Módulo para envío de programación a estaciones repetidoras.** Participé desarrollando aplicaciones para mejorar los procesos de carga de pauta planeada en las estaciones de retransmisión local entre las cuales se encuentran mecanismos de respaldo para el envío de la programación en caso de falla. Los procesos se ejecutan automáticamente todos los días por la madrugada sin necesidad de supervisión manual y cuentan con un módulo de verificación remota.
- **Módulo de generación de reportes de operación.** Desarrollé mejoras al proceso y aplicaciones que se encargan de generar diariamente los reportes de transmisión a nivel nacional de todas las antenas transmisoras donde se encuentra instalado el sistema automático de inserción de comerciales. Optimicé consultas a base de datos y desarrollé la automatización de procesos de recopilación de información desde las estaciones remotas.
- **Subsistema de alertas.** Desarrollé aplicaciones para enviar alertas de fallos en el sistema de inserción de comercialización vía correo electrónico y mensajes SMS al personal de soporte. Algunos de los módulos pueden tomar acciones correctivas en automático para salvaguardar la operación del sistema.
- **Timemachine.** Dentro de la industria televisa existen desarrollos que permiten abrir una ventana de tiempo en la señal nacional dentro de estaciones de retransmisión con el objetivo de insertar más comercialización aún cuando el tiempo disponible planeado ya no lo permite. Este tipo de soluciones descartan

información de audio y video sin que el espectador se dé cuenta, de manera que se abre una pequeña ventana de tiempo para insertar más comercialización. Después de un tiempo el sistema se recupera y deja de descartar información sincronizando la señal nacional recibida con la señal generada en la estación de retransmisión. Realicé las investigaciones necesarias, diseñé, desarrollé e implemente este módulo en ciudades del interior de la república.

- **Procesos de recodificación y entrega automáticos.** La cantidad de archivos digitales con comercialización a transportarse dentro de una televisora es muy grande, algunos archivos pueden incluso medir 2 GB por cada 10 minutos de duración de material. Por lo anterior se requieren procesos automáticos que realicen las diferentes transformaciones entre los formatos propios de las tareas de edición y los formatos utilizados en la transmisión así como almacenamiento final, los cuales tienen una mayor compresión. Este tipo de procesos también se encargan de transportar los materiales mediante redes de datos a servidores de almacenamiento desde donde el sistema los toma para su transmisión. Las actividades que realicé en estos módulos fueron el análisis de requerimientos, diseño, desarrollo, pruebas e implementación.
- **Módulo de diferido.** Una de las operaciones más comunes dentro de la industria televisiva es grabar un programa y transmitirlo más tarde, a esto se le llama un programa diferido. Esta operación puede ser requerida en canales de televisión de paga o bien en canales locales donde se encuentran en otro huso horario. Desarrollé una versión de este módulo, incluyendo la grabación y reproducción.
- **Normalización de audio.** Otro problema que se enfrenta dentro de la industria de la televisión es el nivel de audio al cual son grabados los materiales. Dependiendo de la fuente el nivel de audio de los distintos contenidos puede ser diverso. El control del nivel de audio es muy importante para garantizar una calidad óptima y no saturar los diferentes equipos involucrados en la transmisión, así como para evitar molestias al televidente por la variación de los niveles. He participado en el diseño y desarrollo de distintos módulos y mecanismos para resolver este problema.
- **Procesamiento de audio.** Con la adopción de las nuevas normas para transmisión de la televisión digital terrestre ha evolucionado también la forma de transmitir el audio, en ocasiones es necesario procesarlo para cambiar el número de canales que tiene. Desarrollé e implementé aplicaciones para el procesamiento de audio que expanden virtualmente el audio estéreo a audio 5.1
- **Módulo de sùpers.** Existen algunos elementos en la pantalla que se superponen a la señal original, este tipo de elementos son llamados sùpers, plectras o cintillos. Aprovechando la experiencia que adquirí en mi formación en cuanto a gráficos, he desarrollado módulos que realizan la inserción de este tipo de materiales. Anteriormente esto se llevaba a cabo con sistemas analógicos. El nuevo módulo es capaz de reproducir un archivo digital y las operaciones de superposición se realizan a nivel de píxeles. Con lo anterior se ha logrado dar un salto cualitativo permitiendo a los editores y anunciantes incluir una gran variedad de efectos visuales que antes no eran posibles en sus animaciones.

- **Generación y análisis de streams de transmisión digital.** La adopción de los nuevos estándares digitales para la transmisión también trajo consigo la necesidad de generar la señal de transmisión bajo el estándar ATSC y por lo tanto es fundamental conocer a fondo las normas de dicho estándar. Llevé a cabo las investigaciones necesarias, desarrollé e implementé módulos para realizar la transmisión de señal de televisión en formato digital. Realicé pruebas de concepto que ayudaron al desarrollo final de los proyectos para la transición de la televisión análoga a la transmisión digital en el denominado apagón analógico.
- **Manejo de Closed Caption.** Desarrollé e implementé aplicaciones para incluir closed caption en la transmisión de señal cumpliendo con las normas necesarias en los estándares tanto NTSC como ATSC para su visualización en los receptores finales de los televidentes.
- **Mejoras generales de aplicaciones.** Realicé mantenimiento en las aplicaciones del sistema corrigiendo errores y soportando nuevas características solicitadas por los usuarios. De hecho cuando inicié mis labores mi actividad principal era el mantenimiento de aplicaciones.

Sistemas de reconocimiento de audio

Una vez que la señal es transmitida al aire es muy importante verificar que los contenidos planeados, especialmente la comercialización, han sido reproducidos correctamente. Para ello realicé una investigación acerca de los métodos existentes para reconocimiento de audio. Diseñé, desarrollé e implementé un sistema que es capaz de reconocer el audio de la comercialización dentro de las grabaciones disponibles de la señal en diferentes localidades para una empresa de televisión. Este sistema es capaz de crear de manera autónoma la base de datos de los materiales que se necesitan identificar y está basado en huellas de audio. El sistema realiza el reconocimiento la comercialización las 24 horas del día con un porcentaje efectividad del 99% en 66 localidades. Esta tasa de efectividad se logró con diferentes optimizaciones a lo largo de años de trabajo. El sistema fue desarrollado en lenguaje de programación C/C++.

Sistema de Grabación de señal de televisión

En la televisión, como en cualquier otra industria, es muy importante conocer y monitorear la calidad del producto generado. En este caso el producto es la señal al aire que los espectadores reciben y en la cual está insertada la comercialización. Una de las maneras más eficaces de realizar lo anterior es grabar la señal en distintos puntos de la cadena de distribución y transmisión. Mediante las grabaciones se puede revisar cualquier punto del tren de transmisión, detectar fallas, tener una evidencia de lo que sucedió tanto casos exitosos como cuando suceden problemas, también permite contar con un sustento para realizar cobros y muchas otras tareas de monitoreo.

Para cubrir esta necesidad se ha desarrollado un sistema que permite grabar la señal en televisoras tanto en su control maestro como en localidades donde estas tienen cobertura que ha sido desplegado tanto en el territorio nacional como otros países. Además de

grabar las señales el sistema tiene la capacidad de transportar los videos y concentrarlos en servidores de almacenamiento. Las grabaciones son accesibles para todos los usuarios que las necesiten. Cabe destacar que todos los componentes del sistema son autónomos y funcionan ininterrumpidamente.

El sistema por su despliegue y relevancia consta de varios módulos entre los cuales he participado en el desarrollo de los siguientes:

- **Módulo de envío.** El módulo de envío de grabaciones ha tenido múltiples versiones desarrolladas por otros programadores, sin embargo todas ellas tuvieron alguna falla por la cual fueron descontinuadas. Diseñé la última versión realizando un análisis de las fortalezas y debilidades de las versiones anteriores, además se simplificó el mecanismo y se optimizaron sus operaciones. Realicé el desarrollo y programación de esta versión logrando por fin una aplicación estable, robusta a fallos y condiciones inestables de la red en lugares remotos, incluye mecanismos de detección de fallas e incluso reinicio de operaciones en casos de fallas graves.
- **Módulos de grabación y compresión.** Existen diferentes tipos de hardware que permiten recibir una señal de televisión para su grabación. Diseñé y desarrollé aplicaciones para realizar la grabación de señales de televisión principalmente en banda base. Las aplicaciones graban la señal y comprimen tanto audio como video en formatos adecuados para su almacenamiento. Funcionan de manera automática todo el tiempo generando archivos de segmentos con una duración configurable.
- **Visores de videos para usuario final.** Realicé el mantenimiento de la aplicación mediante la cual los usuarios finales tienen acceso a las grabaciones que realiza todo el sistema. Desarrollé el soporte de nuevas características como búsquedas eficientes, despliegues de diferentes vistas de la información, soporte para nuevos formatos, entre otras. También diseñé y supervisé la migración de la versión bajo ambiente Windows a una versión web del visor. Lo anterior ha generado ventajas como la centralización de la operación, facilidad de instalación para el usuario final, soporte multiplataforma, aumento en la velocidad del tiempo de respuesta en las cargas de video. El visor en versión web se desarrolló utilizando HTML5 y lenguaje de programación PHP, la programación la llevaron a cabo miembros de mi equipo de desarrollo.
- **Diseño de bases de datos.** Realicé el diseño de bases de datos y tablas para almacenar la información de los videos grabados así como de las localidades donde se está grabando la señal. Existía anteriormente una base de datos pero en muchas de las tablas se guardaba información repetida, lo cual ocasionaba que la cantidad de información ocupara mucho espacio en poco tiempo. Lo anterior fue optimizado con una base de datos diseñada para cumplir las 5 formas normales.
- **Módulos de alertas y monitoreo de envío de videos.** La mayor parte de los sistemas que he desarrollado deben ser autónomos y de alta confiabilidad, sin embargo debido a diversas situaciones fuera del alcance de las aplicaciones pueden existir condiciones que impiden sus tareas. Por ello desarrollé aplicaciones automáticas para el monitoreo del envío de las grabaciones de todos los sitios

remotos así como para verificar los parámetros de los archivos incluyendo sus duraciones. Además diseñé y coordiné la programación e implementación de un sistema de alertas que pueda enviar un correo electrónico o bien un mensaje SMS al personal de soporte en caso de que se detecte un problema.

Creación de frameworks

La televisión es un entorno en constante cambio y evolución, como en muchas otras industrias es muy importante responder a las necesidades emergentes con la mayor rapidez posible y así garantizar la competitividad global de la empresa. Los equipos de desarrollo necesitan entonces bibliotecas con código sólido, estable y robusto para que sea la base de los desarrollos sin necesidad de programar todo desde la nada. He participado en el diseño y programación de frameworks que incluyen funciones, métodos y clases utilizados por todo el equipo de desarrollo para facilitar y agilizar la creación de nuevas aplicaciones.

Los frameworks incluyen protocolos de comunicación, métodos de encriptación, manejo de archivos de configuración, manejo de bitácoras de los sistemas, operaciones sobre consultas a disco duro y hardware, clases dedicadas a la conectividad de las aplicaciones con las bases de datos, transmisión de datos mediante multicast, utilerías para obtener parámetros de los archivos digitales de video, clases para el manejo de hardware especializado donde el programador ya no se preocupa de la comunicación directa con los drivers, bibliotecas para realizar conexiones FTP y SFTP, entre otras.

Muchas veces se escribe código que realiza operaciones básicas que pueden ser útiles para varios componentes de un sistema y si se diseña adecuadamente los programadores de esos otros componentes no tienen que preocuparse por implementar funciones que ya han sido desarrolladas. He desarrollado también herramientas tales como un servidor que se encarga de enviar correos electrónicos provenientes de cualquier aplicación, de esta forma el programador que necesita esta característica ya no requiere integrar el protocolo SMTP sino solamente hacer una petición a este servidor, este sólo es un ejemplo pero también he participado en la programación de otras tareas multiuso.

Definición y modificación de procesos

Los sistemas de cómputo surgen para atender una necesidad que tiene cierto grupo de usuarios, áreas o empresas completas. En muchas ocasiones las necesidades que se atienden o automatizan son parte de un proceso mayor que involucra otro conjunto de tareas e incluso interacción con otras áreas. También puede suceder que las aplicaciones deban interactuar con otros sistemas de diferentes equipos de desarrollo, los cuales esperan los resultados de las mismas para realizar sus procesos. He participado en la revisión de los procesos que involucran aplicaciones nuevas a desarrollar o bien en aquéllas en las que se detectan áreas de oportunidad. Actualmente es de vital importancia contar con información oportuna, confiable y de la forma más rápida posible para enfrentar los retos que diariamente se presentan. Participé en el diseño de nuevos procesos que involucran interacciones de los sistemas de desarrollo propio con algunos

desarrollos de otras áreas buscando la mejor solución. Lo anterior es algo muy interesante ya que se tiene contacto con áreas en las cuales los involucrados pueden ser profesionales de distintas disciplinas incluso ajenas a la ingeniería, en estos casos es muy importante saber expresar las ideas claramente y en un lenguaje entendible.

Adopción de sistemas de Help Desk

Debido a la cantidad de aplicaciones desplegadas y el número de usuarios que los sistemas desarrollados tienen, así como un constante crecimiento de despliegue, se ha incrementado también el número de dudas y problemas que tienen los usuarios así como la atención necesaria para los mismos. Coordiné y supervisé con mi equipo de trabajo la instalación de un sistema de Help Desk que permite levantar una petición de soporte, dar seguimiento a cada petición y asignar las tareas a resolver a los integrantes del equipo de soporte. Esta tarea no ha sido de desarrollo sino de investigación, coordinación y administración de una herramienta que fue adquirida a un proveedor externo. Actualmente hemos logrado que todos los reportes se atiendan mediante este portal web el cual además incluye un chat.

Capacitación y supervisión de personal

Dentro de otro grupo de tareas como líder de proyecto capacité al nuevo personal que se integra a mi equipo de trabajo en el conocimiento de las aplicaciones desarrolladas en el área, uso de herramientas especializadas, manejo de los frameworks propios y entornos de trabajo, la búsqueda de nuevas tecnologías y adopción de metodologías de desarrollo de software con los que se busca hacer más eficiente el equipo además de generar software con la mayor calidad posible. Otras de mis actividades consisten en asignar las tareas de programación que hace cada integrante de mi equipo de trabajo de así como vigilar el cumplimiento en tiempo y forma de las mismas.

Reorganización y supervisión del área de soporte

Debido el crecimiento en los últimos años de los proyectos a cargo del Departamento de Desarrollo Tecnológico y su despliegue en todo el país fue necesario reorganizar el área de soporte para brindar una mejor atención. Actualmente me encargo tanto de la supervisión de la sección de desarrollo como de la sección de soporte contando en esta última con un equipo de 10 ingenieros, los cuales atienden cualquier problema que se presente las 24 horas del día los 365 días del año. Como resultado de estas acciones la efectividad de los sistemas de inserción de comerciales nacionales ha alcanzado el 99.5% de éxito según las estadísticas semanales del último año.

Capítulo 3. Desarrollo de un sistema de automatización de máster para televisión

3.1 Breve historia de la televisión

La televisión es un sistema para la transmisión y recepción a distancia de imágenes y sonido que simulan movimiento, para lo cual se emplea un mecanismo de difusión. La transmisión puede ser efectuada mediante ondas de radio, redes de datos, señales digitales o flujos de datos vía satélite, entre otras.

La palabra televisión es un híbrido entre el vocablo griego tele que significa lejos o a distancia y el vocablo latino visionem que podemos entender como visión, desde su raíz etimológica podríamos entender el término como observar a distancia. La televisión hoy en día es uno de los medios de comunicación con mayor alcance y cobertura alrededor del mundo logrando captar diariamente la atención de millones de personas, es accesible de manera pública sin ningún costo mediante la televisión abierta y también existen servicios de pago. La televisión forma parte de la vida cotidiana moderna de la mayor parte de las personas alrededor del mundo.

La historia de la televisión se remonta a finales del siglo XIX época en la cual se realizaron los primeros desarrollos utilizando sistemas mecánicos para realizar la reproducción de imágenes con movimientos tales como el Disco de Nipkow en 1884, el cual “se basaba en un disco metálico con una serie de agujeros cuadrangulares dispuestos en espiral. Al girar, cada agujero recibía una parte de la luz reflejada por la escena que estaba siendo “filmada”. Se trata del primer dispositivo capaz de convertir una imagen en una serie de impulsos eléctricos que, en 1927, serviría de base para los comienzos de la televisión electromecánica” (Palazzesi, 2010) . A principios del siglo XX se iniciaron los desarrollos para utilizar sistemas electromecánicos y electrónicos, gracias a estos avances las primeras emisiones públicas de televisión fueron realizadas por la BBC One en Inglaterra en 1936, en Francia por la TF1 en 1935, mientras que en Estados Unidos la CBS y NBC lograron realizarlo en 1930. En aquél entonces aún se utilizaban gran parte de sistemas mecánicos y no había una programación con horarios regulares como la conocemos hoy en día, solamente se emitían contenidos en ciertas horas del día.

En Alemania durante los años 30's se llevó a cabo el desarrollo que incluía ya un tubo de rayos catódicos, una de las primeras tecnologías electrónicas análogas incluso a la cual se deben muchas características de la televisión moderna, con estos desarrollos durante los Juegos Olímpicos de 1936 se logró realizar la transmisión en directo de los diferentes eventos de la justa deportiva a lugares públicos de toda Alemania. Entre 1936 y 1939 se iniciaron las transmisiones basadas en una programación regular tanto en Inglaterra como en Estados Unidos. Sin embargo durante la Segunda Guerra Mundial las emisiones programadas se vieron interrumpidas reanudándose al final del conflicto bélico.

En 1937 iniciaron las primeras transmisiones basadas en tecnología electrónica gracias a las mejoras en los tubos de rayos catódicos y el iconoscopio en Francia y Reino Unido. En esos años los receptores de televisión contaban con una pantalla muy pequeña y eran muy costosos, sin embargo el número de espectadores empezó a aumentar rápidamente mostrando el gran potencial que tendría la industria de la televisión en su expansión futura.

Durante esta primera era de la televisión las tecnologías eran análogas, la señal de video era una señal eléctrica en la variaba la tensión dependiendo de la luminosidad de la imagen a transmitir y mecanismos de barrido permitían dibujar la imagen que era monocromática. En esta etapa se realizaron estudios acerca de la naturaleza de la visión humana para determinar el número mínimo de imágenes por segundo que se deberían transmitir para que el televidente observe un movimiento fluido, se determinó un mínimo de 24 imágenes por segundo. Este resultado de las investigaciones y otros más de los desarrollos tempranos de la industria constituyen las bases de la televisión moderna.

Después de la primera mitad del siglo XX se empiezan a desarrollar en varias naciones sistemas de televisión públicos y privados, aprovechando el desarrollo tecnológico. Iniciaron grandes avances en el campo de la producción tales como el tener varias fuentes de imagen y alternar entre ellas, la capacidad de las cámaras para realizar acercamientos, las primeras unidades móviles, efectos especiales rudimentarios, incrustación de imágenes sobre la señal de los estudios, etc. Lo anterior dio como resultado la creación de contenidos más atractivos para las audiencias.

También surgieron varios desarrollos para tecnologías de televisión a color, llegando incluso México a contar con un desarrollo propio de esta tecnología con las invenciones del ingeniero Guillermo González Camarena y su sistema de televisión a color el cual patentó en 1940. En México la primera estación de televisión fue el Canal 5 de la Ciudad de México, la cual transmitió por primera vez en 1946. En agosto de 1950 iniciaron las transmisiones de televisión comercial y con ellos los programas regulares. En 1955 se fundó el Telesistema Mexicano, el cual se convertiría con el paso del tiempo en la empresa que hoy conocemos como Televisa.

Los avances de las tecnologías de televisión a color y su implementación trajeron el desafío de crear sistemas que fueran compatibles también con los receptores en blanco y negro que eran los de uso más extendido entre los consumidores. Fue por ello que se crearon conceptos como la luminancia y la crominancia. La luminancia es la información de la imagen que se refiere al brillo de la imagen, lo que corresponde a la imagen vista prácticamente en blanco y negro. La crominancia en cambio es la información relativa a los colores de la imagen.

El desarrollo de la tecnología fue creando regiones de adopción de ciertas metodologías en países vecinos alrededor del mundo llegando a tener desde entonces y hasta la actualidad regiones conformadas de países que utilizan las mismas tecnologías. En Norteamérica se diseñó y adoptó el estándar NTSC y en Europa el estándar PAL. Los

estándares compartían algunas bases e ideas comunes, aunque cada uno contenía diferencias importantes respecto al otro.

A finales de la década de los 80 se empezaron a aprovechar en la televisión las tecnologías digitales incluidas las resultantes del avance de la computación. Se desarrollaron sistemas de digitalización para las tareas de producción y postproducción así como para la transmisión de las señales.

En la década de 1990 se inició el desarrollo de los sistemas de televisión de alta definición en los cuales se incluyeron imágenes con una resolución más alta y una relación de aspecto más agradable a la vista del espectador. En el campo de la alta definición existieron desarrollos análogos en un principio, aunque debido al avance de la computación y los sistemas digitales éstos últimos prevalecieron. Hoy en día todos los sistemas de transmisión son digitales.

Las señales que antes eran análogas empezaron a ser convertidas a una representación en bits donde la información acerca de la luminosidad y el color de la imagen eran almacenadas para ser transportadas. Aparecieron también los algoritmos de compresión de imágenes derivados de la aplicación de la transformada discreta de Fourier, lo anterior permitió disminuir la cantidad de información necesaria tanto para almacenar como transmitir las señales y construir equipos de transmisión aún más económicos y eficientes.

En el campo de la transmisión se empezó a aprovechar la compresión en formato MPEG-2 que permite reducir la cantidad de información necesaria para almacenar el audio y las imágenes en movimiento conservando una buena calidad, aunque es un algoritmo con pérdidas de información que no se pueden recuperar al descomprimir la señal.

Las tecnologías en el campo de la computación empezaron a ser lo suficientemente veloces y con la capacidad de cálculo necesaria para realizar la transmisión, edición y almacenamiento de los diferentes contenidos, la aplicación de sistemas de cómputo se extendió rápidamente en muchos campos de aplicación dentro de la industria de la televisión. Derivado de lo anterior y como resultado de los avances tanto en hardware y software, así como la capacidad creciente de los equipos de cómputo surgieron sistemas de producción en los cuales la señal de televisión es procesada como cualquier otro tipo de información.

Los sistemas de almacenamiento fueron sustituidos por servidores de datos con discos duros donde se puede almacenar una gran de información, en otras palabras pueden almacenar cientos o miles de horas material de video dependiendo del formato en el que esté grabado. Además nació el software de edición que permite realizar producciones mucho más complejas además de permitir una gran gama de efectos en el tratamiento de imagen y sonido a disposición del personal creativo.

Gracias también a la evolución de la televisión en el siglo pasado se consolidaron grandes marcas a nivel mundial que son los proveedores de este tipo de tecnología, generalmente la adquisición de todo este tipo de equipos requiere una gran inversión monetaria así

como su desarrollo y soporte, ya que estos dos elementos son ofrecidos con costos no incluidos en la compra de los equipos.

La industria de la televisión siempre está en constante evolución debido al gran impacto que tiene como medio de comunicación y las demandas de contenido de los espectadores alrededor del mundo. Actualmente en nuestro país se ha sustituido la transmisión analógica por la transmisión digital terrestre en las televisoras abiertas, lo que constituyó el llamado “apagón analógico”. Por otro lado es una industria sujeta a una serie de regulaciones gubernamentales que deben ser cabalmente cumplidas para poder seguir conservando los permisos necesarios para efectuar transmisiones.

La televisión requiere actividades que se realizan las 24 horas del día los 365 días del año, tiene una operación ininterrumpida ya que siempre hay algo en pantalla. Como resultado de la adopción de tecnologías digitales como los servidores de almacenamiento de videos y otros sistemas de cómputo se han logrado crear sistemas de automatización de reproducción, ya sean totalmente automáticos o semiautomáticos. Este tipo de sistemas se basan en la creación de listas de reproducción, denominadas guías de programación dentro del lenguaje de la industria, de manera que los sistemas se encargan de realizar la reproducción de los materiales en el orden que viene establecido así como muchas otras funciones que antes requerían operación manual. La evolución de la televisión de la mano del aprovechamiento de avances en la computación ha generado muchos otros sistemas de cómputo que hacen tareas muy diversas dentro de la industria.

3.2 Sistemas de automatización de máster para televisión

Un sistema de automatización máster para televisión es un conjunto de software e infraestructura que incluye hardware especializado para la emisión de la señal de televisión, equipo de cómputo e incluso equipo de redes de datos. Estos componentes trabajan en conjunto para realizar una reproducción automatizada y confiable de una lista de reproducción que representa la programación de un canal de televisión garantizando la continuidad de la misma así como un entorno de trabajo que permita realizar procesos que van desde la entrega de los materiales audiovisuales procedentes de las tareas de edición hasta la generación de la señal de televisión. Estos sistemas también se conocen como sistemas de automatización de MCR (Master Control).

Los sistemas de automatización de máster constituyen uno de los puntos más críticos en la operación moderna de un canal de televisión debido a que se encargan de generar la señal todo el tiempo.

Existen empresas alrededor de todo el mundo dedicadas al desarrollo de este tipo de productos, las cuales están concentradas en Europa y Estados Unidos. El establecimiento de dichas empresas en tales puntos geográficos se debe a que en estas zonas se han creado los estándares que utiliza la industria de la televisión en nuestro país. Generalmente para implementar las soluciones estas empresas utilizan productos de

diferentes fabricantes de hardware incluyendo componentes que no necesariamente son de su manufactura, de hecho en muchos casos el software tiene la capacidad de funcionar con varios modelos de un componente para cumplir con sus tareas e incluso soporte para varios fabricantes. Lo anterior tiene que ver con la inversión que la televisora decida realizar en la implementación de un sistema de automatización de máster y el costo – beneficio que tenga proyectado.

El esquema de ventas de algunas empresas desarrolladoras de este tipo de productos puede incluir tanto el software como el hardware, o solamente el software a ser utilizado con el hardware que disponga el cliente.

En México las televisoras generalmente adquieren este tipo de productos con empresas extranjeras, las cuales tienen sus oficinas tanto de ventas como de desarrollo fuera del país. Existen muy pocos casos en las cuales existen representantes de ventas directamente en el país, sin embargo también en esos casos los desarrolladores se ubican en el extranjero. Aunque existen casos de éxito en algunos otros países de América Latina como es el caso de Uruguay en donde, como describe Iacobone (2012), la empresa Saeta TV Canal 10 implementó su propio sistema de automatización de máster en el 2012, esta televisora en particular cuenta con un departamento de investigación dedicado a dar soluciones a las necesidades de su canal y cumpliendo con la política de la empresa de diseñar sus propios sistemas.

La cantidad de infraestructura o equipos de cómputo que se destina a este tipo de sistemas es variable, es decir, existen sistemas de automatización donde un mismo equipo de cómputo tiene la suficiente capacidad para generar de 1 a 6 canales de televisión de manera simultánea, y se apoya en más equipos de cómputo para realizar otras tareas, hasta los llamados “Channel in a Box” que consisten en un solo equipo de cómputo que ejecuta todos los módulos que se requieren sin necesidad de apoyarse en otros equipos claro con el riesgo de operación que esto conlleva pero son soluciones mucho más accesibles.

En cuanto a la plataforma de software sobre la cual se ejecutan estos sistemas existen desarrollos sobre sistemas operativos de la familia Windows, otros sobre distribuciones de Linux generalmente descendientes de Debian y algunas otras soluciones implementan incluso sus propios sistemas operativos. Algunos fabricantes despliegan equipos de cómputo que dependiendo de las funciones utilizan un sistema operativo distinto.

Los sistemas de automatización de máster tienen la capacidad de resguardar el material hasta que es necesario para su transmisión contando tanto con el material audiovisual como con metadatos almacenados generalmente en bases de datos, lo cual en la industria recibe el nombre de Media Asset Management o MAM como describe Latin Press (2008).

Dado que son sistemas críticos para un canal de televisión deben manejar algún tipo de redundancia lo cual se lleva a cabo generalmente mediante interfaces de control serial a los equipos de conmutación, aunque también existen interfaces para los mismos mediante

redes IP, sin embargo los fabricantes recomiendan las interfaces seriales por su alta confiabilidad. Otra característica interesante que contemplan son herramientas de autodiagnóstico mediante las cuales pueden aplicar acciones correctivas o bien reportar de forma automática fallas en alguno de sus componentes.

Estos sistemas se plantean para responder ágilmente a los procesos y modificaciones en las listas de reproducción. Como lo detalla el artículo de Latin Press(2008), lo anterior se acentúa más en América Latina por los mercados que manejan y en los canales dedicados a noticias. En América Latina existen particularmente prácticas en las televisoras que incluyen la necesidad de enviar contenidos directamente desde las salas de post-producción a los sistemas de automatización. La tendencia es aumentar la flexibilidad para responder a todo tipo de eventos de una forma dinámica incluso llegando a tener interfaces con redes sociales para desplegar contenidos de las mismas en pantalla y así lograr una mayor interacción entre el público y la señal que se está generando. De hecho en el futuro inmediato de los sistemas de automatización de máster se vislumbra que sea posible operarlos remotamente mediante servicios en la nube.

Por otro lado, en el artículo de Newsline Report (2018) se señala que la adquisición e implementación de un sistema de automatización de máster se considera en las televisoras como una compra estratégica a largo plazo, en la cual se debe evaluar el producto profundamente. La industria de la televisión está estrechamente ligada hoy en día a la tecnología de tal forma que el sistema de automatización de máster que se desee implementar debe responder adecuadamente flujos de trabajo. Uno de los puntos fundamentales a nivel económico para tomar una decisión acerca de qué sistema adquirir es el retorno de la inversión realizada. Un sistema de automatización optimiza y garantiza los ingresos hoy en día de una televisoras.

Los procesos que abarcan los sistemas de automatización incluyen la ingesta de nuevo material, búsqueda de ese material, transcodificación de archivos a diferentes formatos y resoluciones, eliminar esfuerzos redundantes e incluso la prevención de errores en la emisión de la señal.

Antes de iniciar el proyecto se realizó una investigación para saber qué características incluyen estos sistemas, a manera de resumen las soluciones comerciales comúnmente ofrecen las siguientes:

- Reproducción de señal con salida en formato HD o en SD.
- Deben contar con soporte para reproducir materiales en cualquiera de los dos formatos, ya sea HD o SD, realizando las conversiones necesarias para desplegarse adecuadamente en la señal de salida.
- Los videos de la lista son archivos de audio y video en formato digital con los principales estándares de la industria (mpeg2, mp4, mov, mxf).
- Inserción de cuetones para realizar tareas de conmutación en los receptores de la señal generando a través de los mismos una interfaz con los sistemas de inserción automática de comercialización regional.
- Interfaz con los sistemas de tráfico para la recepción de guías planeadas.

- Posibilidad de realizar cambios en la lista de reproducción respondiendo de manera ágil a cambios incluso de último momento.
- Generación de informes de reproducción para conciliar los materiales transmitidos en los sistemas de tráfico, permitiendo así realizar los cobros correspondientes a los anunciantes teniendo la seguridad de que el material se presentó al aire en la señal emitida.
- Base de datos para almacenar los metadatos de los materiales audiovisuales.
- Vista previa de los materiales para asegurar la calidad de los mismos y que el contenido es el correcto.
- Reloj de cuenta regresiva en el reproductor que permita saber exactamente cuánto tiempo falta para salir a corte comercial o bien regresar de bloque comercial a programa.
- Inserción de logos en la señal para dar identidad al canal de televisión a través de gráficos.
- Generador de caracteres para incluir avisos en la señal de noticias importantes, notas urgentes, emergencias o cualquier otra información. Generalmente las leyendas contienen algún tipo de animación.
- Opción de insertar señales en vivo provenientes de estudios alternando de ser necesarias con fuentes grabadas y comercialización.
- Gestión de contenidos grabados, incluyendo almacenamiento y clasificación.
- Inserción de gráficos en tiempo real para dar identidad al canal como parte de su imagen comercial o bien ofrecer productos a los anunciantes para incorporar publicidad sobre programas televisivos.
- Generación de la señal de salida en SDI (banda base) o en un stream de datos IP.
- Efectos de transición de video permitiendo realizar animaciones al cambiar entre fuentes o pasar a bloque comercial.
- Control de volumen para ajustar el nivel del audio en la señal de salida.

La investigación realizada nos permitió identificar que para soportar las características antes descritas estos sistemas, en su mayoría, cuentan con un diseño modular en el cual se identificaron los siguientes componentes:

- Módulo de Ingestión. Es un módulo encargado de la recepción de los materiales audiovisuales provenientes de edición y el ingreso de los mismos al sistema registrando sus metadatos en la base de datos del sistema.
- Módulo de preparación de la guía. Este módulo puede estar integrado en el sistema de automatización, su función es permitir a los usuarios planear con anterioridad a la fecha de transmisión qué materiales y en qué orden serán reproducidos. Otra modalidad puede ser una interfaz al sistema de tráfico en el cual se realizan esas tareas. Los sistemas de tráfico manejan más información como los contratos comerciales con los anunciantes, información legal, control de inventarios y otros elementos que los hacen complejos, por lo que generalmente son sistemas independientes de los sistemas de automatización de máster.

- Módulo de almacenamiento. Es la parte del sistema dedicada a almacenar de una forma ordenada y clasificada los materiales para ser reproducidos. Generalmente cuenta con una base de datos para guardar los metadatos de los materiales audiovisuales y servidores con una gran capacidad de almacenamiento para contener semanas e incluso meses de programas entregados con anterioridad a su transmisión. Estos servidores también son llamados videoservidores.
- Módulo de reproducción. El reproductor del sistema que genera la señal de televisión con base a las listas de reproducción preparadas con anterioridad. En este se pueden realizar las modificaciones a la lista de reproducción así como hacer cambios o agregados de último momento. Otras de las características que incluye este módulo son la conmutación a fuentes externas de video, la inserción de cuetones, gráficos y logos. Este módulo es el encargado de la generación de la señal de salida del sistema ya sea en banda base, stream IP o alguna otra forma de transmisión. Una característica muy importante con la cual debe contar el módulo de reproducción es un esquema de redundancia de manera que se garantice todo el tiempo la emisión de la señal. El módulo de reproducción generalmente se le conoce también como PlayOut.
- Módulo de control de switchers: Algunos sistemas de automatización también cuentan con capacidad para controlar hardware externo en el que podemos seleccionar la fuente de la señal que se encuentre al aire o bien conmutar manualmente a un equipo de respaldo.
- Módulo de streaming: En el caso que el reproductor tenga la capacidad de entregar la señal en algún formato digital para ser transmitido mediante una red de datos existe un módulo encargado de realizar la codificación y el tratamiento adecuado de la señal para convertirlo en paquetes de información que serán enviados mediante las redes de datos bajo los protocolos de comunicación necesarios para ser recibidos en otro equipo. En algunos casos los reproductores pueden generar tanto la señal en banda base como el stream en IP.
- Módulo de monitoreo. Una sección del sistema que nos permite ver la señal de salida para asegurarnos que todo está funcionando adecuadamente. En algunas soluciones también contiene alguna información de diagnóstico que puede ser enviada en automático a los equipos de soporte técnico.
- Módulo de entrega de reportes de reproducción. Una vez que una lista de reproducción ha sido reproducida en su totalidad se generan reportes para saber exactamente en qué momento fueron reproducidos los materiales, incluyendo todas las modificaciones que se hayan realizado a la lista de reproducción en el sistema. Estos reportes pueden ser entregados al sistema de tráfico en el cual se planearon las listas de reproducción o guías de programación con la finalidad de

comparar lo planeado con la señal que se emitió al aire. Los reportes generados son muy importantes, deben reflejar de forma exacta la hora real de transmisión de cada material porque serán la base para realizar los cobros a los anunciantes, lo que representa una gran fuente de ingresos para las empresas de televisión.

- Módulo de generador de caracteres. En este tipo de módulos permiten incrustar letreros en movimiento con diferentes diseños y animaciones sobre la señal para anunciar eventos importantes, noticias, horarios etc.

3.3 Objetivo del proyecto

El objetivo del proyecto fue crear un sistema de automatización de máster a la medida para emitir la señal de los canales de una televisora utilizando la infraestructura de redes y bases de datos disponibles en la misma así como aprovechar el hardware disponible para la generación y adquisición de señales. El sistema de automatización de máster debería realizar la reproducción de una guía de programación continua los 365 días del año y las 24 horas del día con alto grado de confiabilidad así como incluir la mayor parte de las características disponibles en los productos comerciales descritos anteriormente.

La creación de un sistema de automatización de máster desarrollado a la medida trae como ventajas una menor inversión en la implementación del mismo aprovechando los recursos que ya se cuentan en hardware, redes y bases de datos. Además se pueden agregar mejoras que nos permiten ampliar las capacidades y características del sistema a la medida de los flujos operativos con que cuente la televisora, en caso contrario si fuera adquirido un producto comercial se elevarían tanto los costos de implementación incluyendo la adquisición de equipo como posteriormente el costo de todas las adecuaciones que a lo largo del tiempo se deban ir realizando. Generalmente las adecuaciones de estos sistemas comerciales son requerimientos especiales por parte del cliente que requieren cierta inversión.

Para el desarrollo de este sistema de automatización de máster fueron solicitados los siguientes requerimientos generales:

- Utilizar la base de datos disponible, la cual ejecuta en un servidor Oracle de Base de Datos.
- Aprovechar el hardware disponible para la emisión y captura de señal en banda base disponible, el cual en ese momento consistía en tarjetas marca Blackmagic modelo DeckLink Quad.
- Los reproductores o playout deben ser diseñados para ejecutar en servidores disponibles de bajo perfil con sistema operativo Windows 7, que fueron los designados para este proyecto.
- Soporte para 4 canales independientes con su propia guía de programación.
- Debería contar con módulos de ingestión mediante los cuales sea posible enviar material desde las cabinas de edición hasta el almacenamiento del sistema. Estos materiales usualmente se generan en archivos encapsulados en formato MOV pero pueden contener audio y video en diferentes codificadores. El video puede ser resolución SD o HD. Los materiales generados son programas y spots. El módulo de ingestión debe contar con la capacidad para que los materiales sean entregados con anterioridad.
- Contar con un módulo de almacenamiento que permita tener catalogados los materiales para la identificación dentro del sistema.
- Una interfaz con el sistema de tráfico de la televisora para poder recibir las guías planeadas.
- Capacidad para generar informes de reproducción al término de la guía de transmisión diariamente.

- Soporte para 3 canales que cuentan con programación 100% grabada, para los cuales se necesitará el manejo de la guía totalmente automatizada.
- Soporte para un canal con la posibilidad de recibir programas en vivo desde un estudio en banda base con resolución HD, los cuales requerirán de la operación de un usuario para enviar la señal a corte comercial.
- Salidas de los reproductores en señal SDI tanto en resolución HD como en SD de manera simultánea, en el caso de la señal SD debe ser una imagen resultante de aplicar “central cut” de la señal HD.
- Capacidad de inserción de logos en los bloques de programa de la señal.
- Inserción de súpers en la señal de salida.
- En programas grabados en resolución SD el reproductor debe insertar “pillar box” para llenar la pantalla en HD.
- Capacidad para insertar cuetones en los canales de audio 7 y 8 de la señal de salida.
- Funcionalidad para diferir contenidos de otras señales correspondientes a la televisora en los canales con el sistema de automatización.
- Posibilidad de importar spots y clips de video incluso de otros países, el mecanismo de de envío no forman parte del alcance de este proyecto.

Con base en estos requerimientos generales realizó el análisis las necesidades de cada uno de los módulos con los cuales debería contar el sistema, y de esta forma sentar las bases del posterior desarrollo e implementación.

Para dar solución a este proyecto y basado en los productos analizados en la industria se propusieron los siguientes módulos.

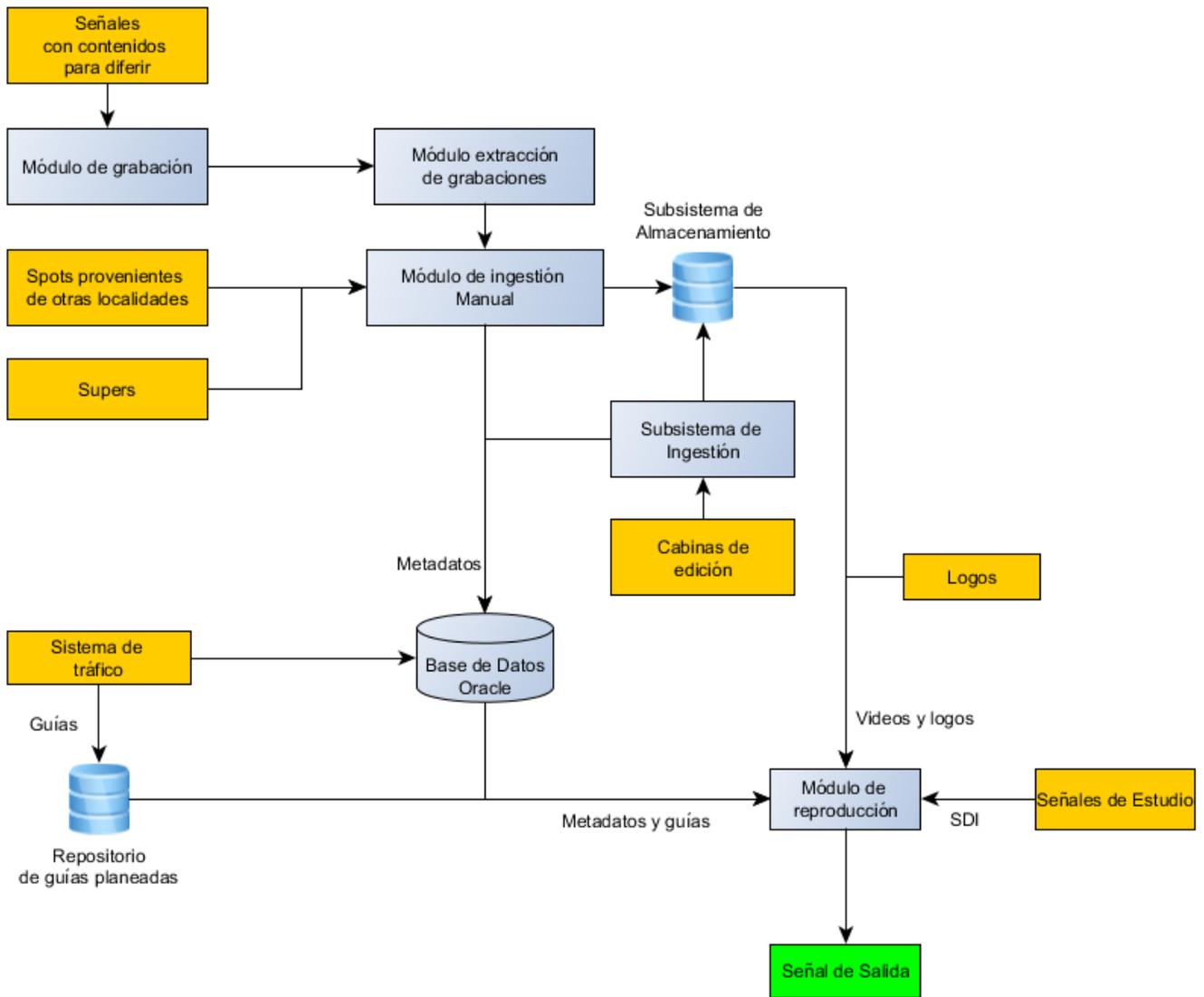


Figura 2 Módulos propuestos para sistema de automatización de máster

3.4 Subsistema de ingestión

El primer subsistema en ser analizado con mayor profundidad fue el subsistema de ingestión. Para ello se realizaron entrevistas con las áreas de edición encargadas de la generación de los contenidos con el propósito de conocer sus necesidades.

Los requerimientos fueron los siguientes:

- Las cabinas de edición funcionaban bajo plataformas MacOS, por lo que el subsistema de ingestión debía ser multiplataforma, ya que no se descartó la posibilidad de que puedan entregar desde plataforma Windows.
- Los archivos provenientes de edición como ya se ha mencionado podían venir encapsulados en formato MOV, tanto en resolución HD como en SD. Los códecs de video podían incluir estándares MPEG2 y H264 a diferente bitrate. En cuanto el audio los archivos podían contener información en códecs PCM, MP3 o bien AAC.
- La operación habitual de las cabinas de edición para la generación de contenidos incluía entregar los programas completos, es decir, generaban un solo archivo de video que contiene todos los bloques de un programa. Entre cada bloque de programa incluían aproximadamente 10 segundos de video en negro y silencio, cuando estos videos eran entregados a los sistemas de automatización con los que contaban los califican indicando los tiempos de inicio y fin de cada bloque.
- La operación del área de edición tenía varios escenarios para la fecha de entrega así como la anticipación con la cual podían entregar los materiales audiovisuales. En un primer escenario los materiales podían ser entregados con meses o semanas de anticipación. En un segundo escenario eran entregados el mismo día de la transmisión. El tercer escenario incluía entregar por separado cada uno de los bloques de programa por separado minutos antes de su transmisión, por lo cual surgió la necesidad de contar con un proceso ágil en este último escenario.
- En cuanto a los spots se manejaban dos escenarios, uno en el cual eran entregados el mismo día de su transmisión o bien con días de anticipación.
- El sistema de almacenamiento debería guardar los materiales hasta que se requieran para su transmisión. Incluso se solicitó la capacidad para almacenar e identificar materiales correspondientes a programas que se pueden retransmitir en otras fechas y hubieran sido entregados con semanas o meses de antelación.

Una vez realizadas las entrevistas para el levantamiento de requerimientos, la fase de análisis de requerimientos no sólo se debe limitar a la atención de las necesidades planteadas por el cliente, un buen análisis de requerimientos debe tomar en cuenta toda la información disponible y con base en esta proponer una solución eficaz y óptima, la cual puede incluir ideas que el cliente originalmente no había planteado pero el hecho de implementarlas da un valor agregado al producto. Por otro lado el tratamiento adecuado de la información nos permite desarrollar no solamente un producto de software sino el mejor producto posible.

El análisis de las historias y necesidades planteadas por el usuario nos llevó a modelar los distintos niveles de la información que manejan.

En el plano más general de la información fue identificado el canal. Un canal manejaba producciones (o programas televisivos) que pueden ser transmitidos dentro de la programación. A su vez cada producción se componía de una serie de programas o capítulos. Y finalmente en el nivel más bajo se identificó que cada capítulo se compone de una serie de bloques entre los cuales en la transmisión se programaban bloques de comerciales.

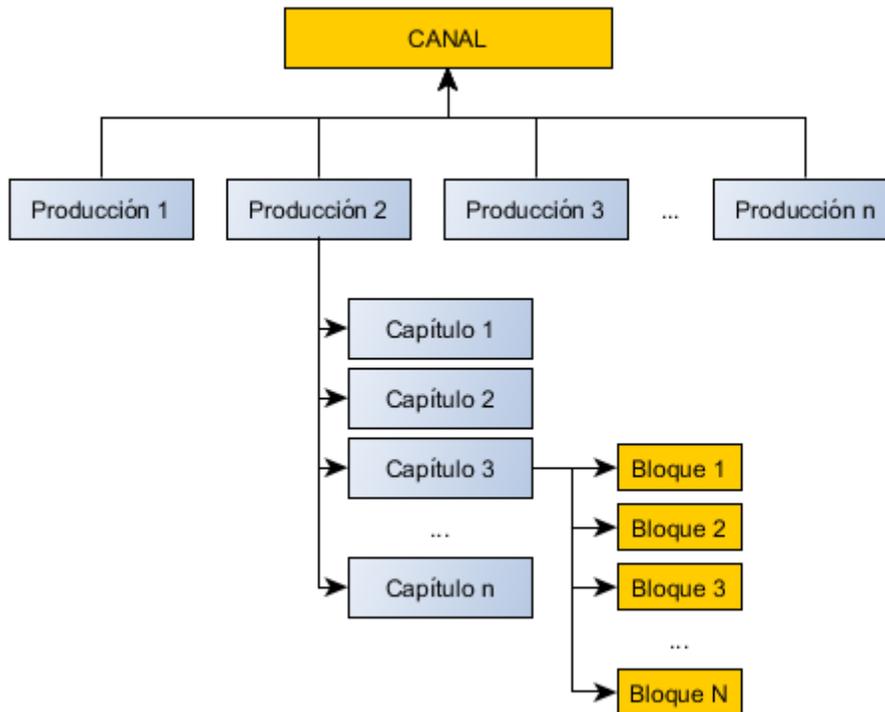


Figura 3 Niveles de información para operación de un canal de televisión

Dentro de las tareas operativas existentes se tenía la noción de esta organización más no estaba definida dentro de los procesos. Una de las primeras propuestas al plantear la solución fue llevar a cabo el registro de todos estos niveles de la información incluyendo los canales, las producciones correspondientes y un número de capítulo para almacenar el material de una manera organizada e identificada de forma inequívoca.

El sistema de tráfico era desarrollado por otra área de sistemas de la empresa y forma parte de otro grupo de soluciones. En ese sistema de tráfico se planeaban y generaban las guías de programación al menos con un día de anticipación. Una guía de programación es una lista de reproducción que contiene el orden de cada uno de los materiales que van a reproducirse durante todo el día. Generalmente estas guías de programación cubrían un horario de 6 de la mañana a las 6 de la mañana del siguiente día. Cada guía de programación contenía entonces la lista de reproducción de todo un día o 24 hrs.

Típicamente la estructura de una guía de programación incluía bloques de programas entre los cuales hay bloques de comerciales. Los bloques de comerciales pueden contener uno o más comerciales y dentro del lenguaje televisivo se conocen como “carriers”.

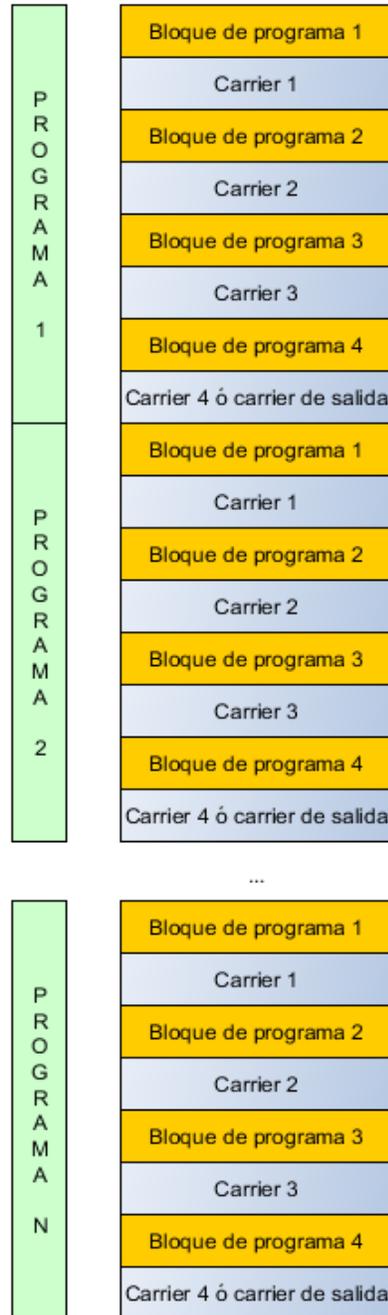


Figura 4 Estructura de una guía de programación

Cada uno de los elementos de la guía, ya sea un bloque de programa o un comercial, recibían el nombre de “evento”. Los eventos tenían varias propiedades acerca de los materiales a reproducir como la fuente del material que puede ser un archivo almacenado en el disco duro o una fuente de video externa proveniente de un estudio, la duración del material, su descripción o nombre y un identificador en el caso de los materiales que se reproducen desde el disco duro. El identificador es esencial ya que se asignaba dentro del sistema al archivo de video correspondiente a un evento dado sin ambigüedad.

En el sistema de tráfico se manejaban estos identificadores y recibían el nombre de código de inventario. Hasta el momento de creación de este sistema de automatización los únicos eventos que tenían asignado un identificador eran los materiales correspondientes a los comerciales, pero los bloques de los programas no.

La colaboración entre el área encargada del desarrollo del sistema de tráfico y el Departamento de Desarrollo Tecnológico siempre ha sido muy estrecha, ya que en una gran cantidad de proyectos se construyeron interfaces entre ambos sistemas. Lo anterior se debe a que en este sistema de tráfico se realiza la planeación de la transmisión y se ofrecía como parte de las soluciones, la información era entregada a los diferentes sistemas de automatización que desarrollamos en el Departamento de Desarrollo Tecnológico para realizar transmisión, al día siguiente de efectuada la transmisión nuestros diferentes sistemas de automatización entregaban los reportes de la transmisión al sistema de tráfico para los propósitos que la televisora cliente los ocupe como lo son el cobro de los comerciales a los anunciantes.

Para el desarrollo de este proyecto se utilizó parte de la experiencia adquirida con base en colaboraciones anteriores así como mecanismos de comunicación ya probados en otras interfaces que han demostrado tener un funcionamiento óptimo. En la actividad diaria de un ingeniero observé a lo largo del tiempo la importancia de saber trabajar en equipo para lograr un objetivo, y este trabajo en equipo incluye tanto colaboraciones con los miembros del área donde uno labora así como trabajo en equipo con otras áreas y la construcción de acuerdos que permitan lograr las metas en tiempo y forma.

Desde el punto de vista del desarrollo del subsistema de ingestión se tomó en cuenta la colaboración e integración de ambos sistemas. Para lograrlo se investigó con el área encargada del sistema de tráfico y encontramos que ellos contaban con una base de datos donde los usuarios dan de alta las producciones televisivas incluso semanas o meses antes de que estos salgan al aire por primera vez. También se tenía que resolver el requerimiento en el cual en el sistema de ingestión deberían poder entregar materiales con semanas de anticipación o incluso meses, pero código de inventario se entregaba a nuestros sistemas solamente a través de las guías de programación el día anterior a la transmisión.

Se analizó entonces la forma de crear un identificador que nos permitiera conocer con exactitud de qué programa, capítulo y producción es un material, en un formato que ambas áreas pudieran determinar bajo las mismas reglas y diera el mismo resultado

independientemente del momento en el que se genere. Para tal objetivo se ideó un identificador de 16 dígitos de longitud constante compuesto de la siguiente manera:

- 8 dígitos para un identificador de programa proveniente de la base de datos del sistema de tráfico. El identificador se rellenó siempre a 8 posiciones colocando 0 a la izquierda si era necesario.
- 2 dígitos para el año de estreno del capítulo, esto nos permitía reiniciar la numeración de capítulo cada año de ser necesario.
- 4 dígitos que representan un número de capítulo iniciando desde el número 1, lo cual nos permitió tener hasta 9999 capítulos de un programa, esto representaba la emisión de 27 años de un programa que se transmite diariamente. Los cuatro dígitos se rellenaban con 0 hasta completar los 4.
- 2 dígitos más para el bloque de programa. Una guía generalmente era una sucesión de bloques de programa y comerciales, dos dígitos nos permitieron la inclusión de hasta 99 bloques de programa aunque típicamente un programa como máximo tenía hasta 10 o 12 bloques. En este caso se rellenaron con 0 hasta completar las 2 posiciones.

La creación de este acuerdo fue un punto primordial que se aprovechó en la ingestión, el almacenamiento de materiales, creación de guías y su carga en el módulo de reproducción. Con este identificador único se podían almacenar todos los datos importantes de un video que era entregado desde las cabinas de edición identificando a qué programa pertenece, qué número de capítulo es y qué bloque es. El identificador descrito es el código de inventario para los bloques de programa. Además se propuso al usuario la numeración de los capítulos, un flujo operativo que nunca había existido en la empresa cliente. El llevar a cabo la numeración de los capítulos de una producción permitió que el usuario al crear la guía en el sistema de tráfico únicamente indicara qué número de capítulo de qué producción requería transmitir, y podía planear la transmisión varias veces en el día utilizando el mismo material o bien en fechas diferentes, logrando la retransmisión de un mismo contenido de forma inequívoca y sin necesidad de tener que volver a entregar el mismo archivo de video con varios identificadores para sus diferentes retransmisiones como se venía realizando.

Tomando en cuenta los requerimientos se realizó un análisis identificando los siguientes componentes para su desarrollo.

- Diseño de la base de datos para el proceso de ingestión de materiales

Módulos para ingestión de programas completos:

- Módulo interfaz WEB de alta de programas.
- Módulo interfaz WEB de alta de capítulos.
- Módulo interfaz WEB para ingestión de programas.
- Módulo de transcodificación para programas.
- Almacenamiento de programas en baja resolución.
- Módulo interfaz WEB para consulta de programas ingestados.

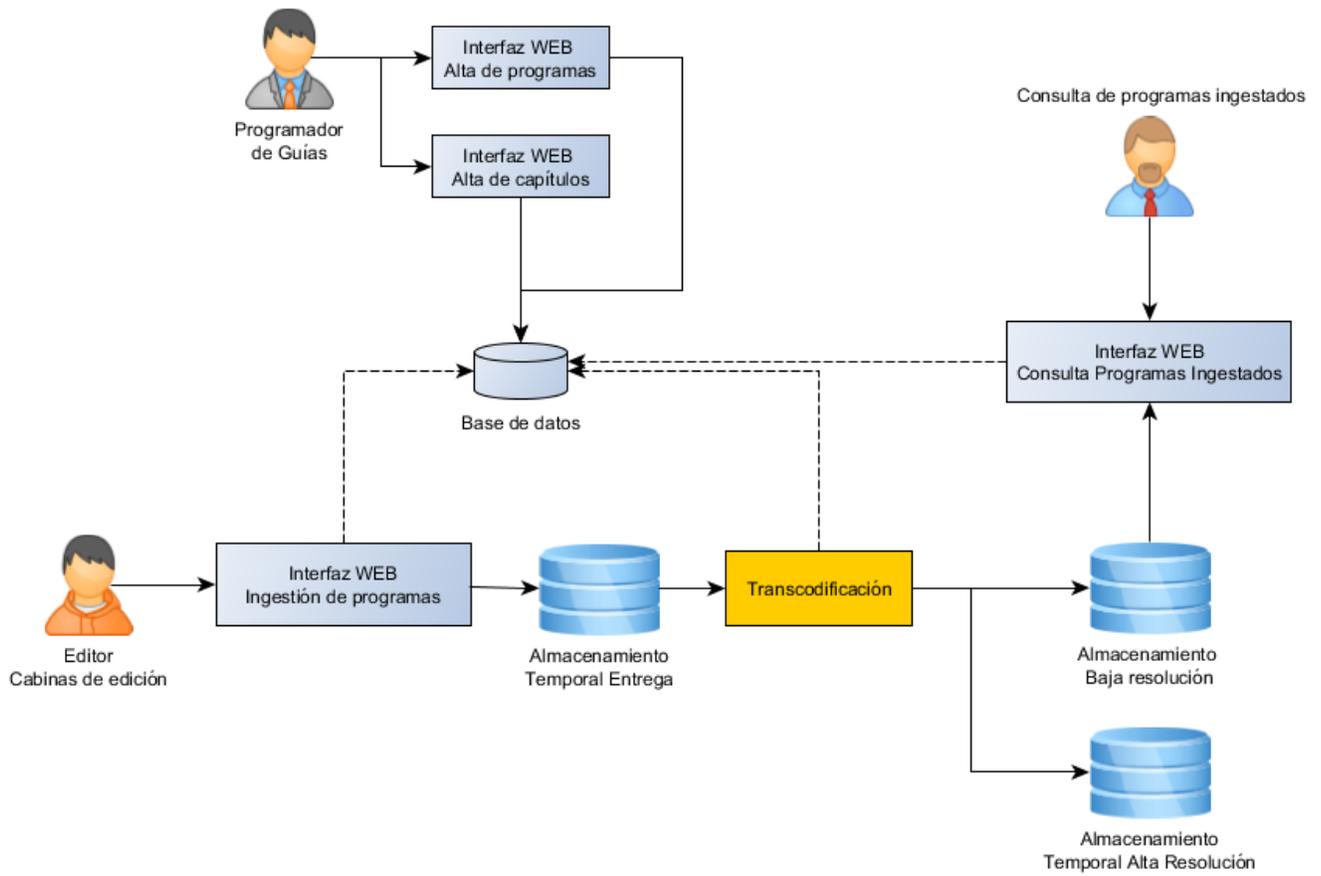


Figura 5 Módulos del subsistema de ingestión para ingreso de programas completos

Módulo para ingestión de spots y bloques individuales:

- Módulo interfaz WEB ingestión de spots y bloques de programa.
- Módulo de transcodificación automática de spots.
- Módulo catálogo de spots ingestados.
- Módulo de Ingestión manual

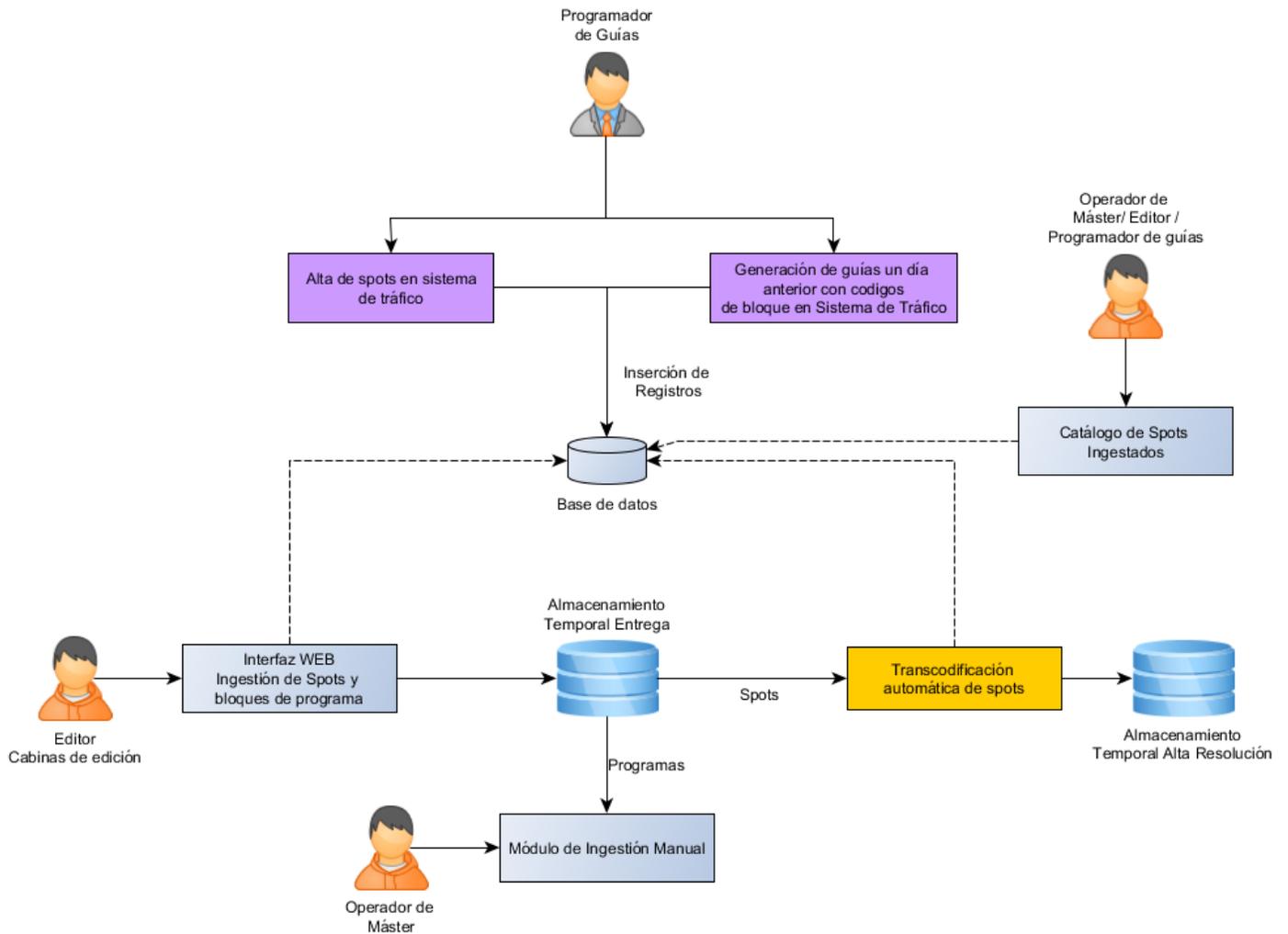


Figura 6 Módulos del subsistema de ingestión para spots y bloques individuales

- Herramientas auxiliares
- Módulo borrado materiales de programas

3.4.1 Diseño de la base de datos para la ingestión de materiales

La base de datos que se empleó para este proyecto está alojada en un servidor de base de datos Oracle. En este servidor se creó un esquema de base de datos el cual contenía las tablas para almacenar la información del sistema de automatización.

Además se creó un usuario aplicativo para que las diferentes aplicaciones accedieran a la base de datos solamente con los permisos necesarios, esto es una buena práctica de seguridad en vez de dar acceso con el usuario propietario de las tablas.

El diseño de la base de datos para la entrega de materiales incluyó las siguientes tablas:

Tabla 1. Diseño de la base de datos para entrega de materiales

Tabla	Propósito
TVP_CANALES	Catálogo de canales disponibles en el sistema de automatización.
TVP_PROGRAMAS	Catálogo de las producciones televisivas registradas en el sistema de automatización.
TVP_CAPITULOS	Catálogo de los capítulos de todas las producciones.
TVP_BLOQUES_PROG	Almacena los bloques de programa de cada capítulo generados
TVP_CODIFICADORES	Catálogo de las aplicaciones encargadas de la transcodificación
TVP_PETICION_REINGES_PROG	Almacena las peticiones para el proceso de borrado de materiales
TVP_CAT_EDO_PET_REINGES	Catálogo de estados de proceso de reingestión
TVP_ALMAC_WEB	Catálogo de discos disponibles en el servidor de baja resolución para mostrar previos de las producciones televisivas enviadas al sistema.

La estructura de cada tabla es la siguiente:

Tabla 2. Estructura tabla base de datos TVP_CANALES

Campo	Tipo	Propósito
ID	Number(4)	ID único para cada canal. Es la llave primaria de la tabla
Nombre	Varchar2(32)	Nombre del canal

Tabla 3. Estructura tabla base de datos TVP_PROGRAMAS

Campo	Tipo	Propósito
ID	Number(8)	ID único para cada programa, es la llave primaria de la tabla
ID_TRAFICO	Number(8)	ID en el sistema de tráfico del programa
Nombre	Varchar2(64)	Nombre del programa
Year	Number(4)	Año de alta del programa
ID_Canal	Number(4)	ID del canal al que corresponde la producción. Llave foránea proveniente de la tabla TVP_CANALES
Enabled	Number(1)	Permite desactivar o activar el programa de todas las vistas del sistema
Depuracion	Number(1)	Permite encender una bandera para saber si el programa debe ser depurado de los servidores de almacenamiento

Tabla 4. Estructura tabla base de datos TVP_CAPITULOS

Campo	Tipo	Propósito
ID	Number(12)	ID único para cada capítulo, es la llave primaria de la tabla
ID_programa	Number(8)	ID del programa al que pertenece el capítulo, llave foránea de la tabla TVP_PROGRAMAS
Num_cap	Number(4)	Número de capítulo dentro de la producción en la que está registrado
Nom_cap	Varchar2(64)	Nombre del capítulo
Year_cap	Number(4)	Año de estreno del capítulo, planeado para reiniciar la cuenta por año si se requiere.

Num_bloq	Number(2)	Número de bloques que tiene el programa
Fecha_tx	Date	Fecha planeada para el estreno del programa
Id_codificador	Number(2)	Id del transcodificador que realizó el proceso, llave foránea proveniente de la tabla TVP_CODIFICADORES
Sub_cod_inv	Varchar(14)	Código de inventario a 14 posiciones, no contiene los dígitos de cada bloque
Duracion	Number(9)	Duración acumulada de todos los bloques del capítulo en milisegundos
Tamano	Number	Tamaño en disco acumulado de todos los bloques del capítulo en bytes
Enviado	Number(2)	Estado del envío al servidor WEB 0=Nuevo 1=Enviando 2=Enviado
Transcodificado	Number(2)	Estado de la transcodificación 0 = Nuevo 1 = En proceso 2 = Transcodificado
Almacenamiento	Number(2)	Estado del envío al almacenamiento final 0 = Nuevo 1 = Enviando 2 = Enviado a subsistema de almacenamiento
Fecha_ingreso	Date	Fecha en la que fue enviado mediante el subsistema de ingestión
Formato_video	Number(2)	Indica el formato de video del capítulo (HD o SD)
Tiempo_ini1	Varchar2(11)	Timecode del inicio del primer bloque
Tiempo_fin1	Varchar2(11)	Timecode del final del primer bloque
Tiempo_ini2	Varchar2(11)	Timecode del inicio del segundo bloque
Tiempo_fin2	Varchar2(11)	Timecode del final del segundo bloque
Tiempo_ini3	Varchar2(11)	Timecode del inicio del tercer bloque

Tiempo_fin3	Varchar2(11)	Timecode del final del tercer bloque
Tiempo_ini4	Varchar2(11)	Timecode del inicio del cuarto bloque
Tiempo_fin4	Varchar2(11)	Timecode del final del cuarto bloque
Tiempo_ini5	Varchar2(11)	Timecode del inicio del quinto bloque
Tiempo_fin5	Varchar2(11)	Timecode del final del quinto bloque
Tiempo_ini6	Varchar2(11)	Timecode del inicio del sexto bloque
Tiempo_fin6	Varchar2(11)	Timecode del final del sexto bloque
Tiempo_ini7	Varchar2(11)	Timecode del inicio del séptimo bloque
Tiempo_fin7	Varchar2(11)	Timecode del final del séptimo bloque
Tiempo_ini8	Varchar2(11)	Timecode del inicio del octavo bloque
Tiempo_fin8	Varchar2(11)	Timecode del final del octavo bloque
Tiempo_ini9	Varchar2(11)	Timecode del inicio del noveno bloque
Tiempo_fin9	Varchar2(11)	Timecode del final del noveno bloque
Tiempo_ini10	Varchar2(11)	Timecode del inicio del décimo bloque
Tiempo_fin10	Varchar2(11)	Timecode del final del décimo bloque
ID_almac_baja	Number(2)	Id llave foránea de la tabla TVP_ALMAC_WEB

Tabla 5. Estructura tabla base de datos TVP_BLOQUES_PROG

Campo	Tipo	Propósito
ID	Number(14)	ID único para cada bloque de programa, es la llave primaria de la tabla
ID_cap	Number(12)	ID del capítulo al que pertenece el bloque de programa, llave foránea de la tabla TVP_CAPITULOS
Num_bloq	Number(2)	Número de bloque dentro del capítulo
Duracion	Number(9)	Duración del bloque con precisión de milisegundos
Duracion_cad	Varchar2(12)	Duración en código de tiempo del bloque, se almacena para no hacer el cálculo en los demás módulos
Tamano	Number	Tamaño en disco del bloque en alta resolución
Tamano_cad	Varchar2(16)	Tamaño en disco del bloque en cadena con el cálculo en

		GB
Codigo_material	Varchar2(12)	Código único del video dentro del sistema de automatización
Recodificado	Number(2)	Bandera que indica si fue transcodificado
Master	Number(2)	Bandera que indica si el bloque ya fue enviado al sistema de almacenamiento
Fecha_ingreso	Date	Fecha en la que el bloque ingresó al sistema

Tabla 6. Estructura tabla base de datos TVP_CODIFICADORES

Campo	Tipo	Propósito
ID	Number(2)	ID único para cada transcodificador, es la llave primaria de la tabla
IP	Varchar2(15)	Ip del equipo donde está instalado el transcodificador
Estado	Number(3)	Estado actual del transcodificador

Tabla 7. Estructura tabla base de datos TVP_PETICION_REINGES_PROG

Campo	Tipo	Propósito
ID	Number(12)	ID único para cada petición de reingestión, es la llave primaria de la tabla
Capitulo	Number(12)	ID del capítulo al que pertenece la petición, llave foránea proveniente de la tabla TVP_CAPITULOS
Fecha_peticion	Date	Fecha en la que se realizó la petición
Estado	Number(2)	Estado de la petición, llave foránea proveniente de la tabla TVP_CAT_EDO_PET_REINGES

Tabla 8. Estructura tabla base de datos TVP_EDO_PET_REINGES

Campo	Tipo	Propósito
ID	Number(2)	ID único para cada estado del catálogo, es la llave primaria de la tabla
EDO_CAD	Varchar2(64)	Nombre del estado en texto

Tabla 9. Estructura tabla base de datos TVP_ALMAC_WEB

Campo	Tipo	Propósito
ID	Number(3)	Id único para cada disco disponible para alojar baja resolución
RUTA_WEB	Varchar2(128)	Ruta del alias WEB para acceder a la ruta de almacenamiento
RUTA_FISICA	Varchar2(128)	Ruta real dentro del servidor

3.4.2 Módulo interfaz WEB de alta de programas

El módulo interfaz WEB de alta de programas permitió a los usuarios registrar producciones televisivas en el sistema de automatización con la finalidad de dar de alta posteriormente capítulos de esas producciones en el sistema y realizar el proceso ingestión de los materiales. Los usuarios de las cabinas de edición desde donde se generan los contenidos utilizaban sistemas operativos MacOS, sin embargo algunos otros empleados, por ejemplo los programadores de las guías y editores de cápsulas comerciales utilizaban sistemas operativos Windows, por ello la mayor parte de las interfaces del sistema de automatización para usuarios finales fueron solicitadas para ser multiplataforma. Contemplando este requerimiento se planteó realizar el desarrollo de las interfaces para usuarios finales con alguna tecnología WEB.

Los recursos disponibles constaban de servidores con sistema operativo Windows 7. Para la programación de estas interfaces se decidió usar el lenguaje de programación PHP del lado del servidor ejecutando en un servidor Apache en plataforma Windows 7. Del lado del cliente se ocupó JavaScript, CSS3 así como HTML5. Esta versión es el último estándar de HTML, sin embargo no todos los navegadores web en el momento del desarrollo eran capaces de soportar todas sus características.

En las pruebas disponibles en línea para validar el soporte de HTML5 el navegador WEB que mejor puntuación obtuvo fue Google Chrome, por lo que fue recomendado a los usuarios para utilizar las interfaces Web del sistema de automatización. El navegador Google Chrome contaba con versiones para la mayor parte de los sistemas operativos por lo que el requerimiento de multiplataforma fue cubierto de esta forma.

Los servidores contaban además con una lista de acceso o ACL que restringe la comunicación permitiendo solamente las conexiones desde los equipos aprobados. La ACL era administrada por el área de redes y telecomunicaciones del cliente.

En este módulo realicé el diseño de la funcionalidad que debería tener y la codificación se encargó a uno de los programadores de mi equipo de trabajo.

El funcionamiento del módulo que definió es el siguiente:

- Mostrar un listado de los diferentes canales que fueron registrados en el sistema de automatización, los cuales están almacenados en el catálogo de canales. Los canales se darían de alta directamente en la base de datos. El usuario selecciona entonces el canal para el cual desea dar de alta la producción.
- Solicita el nombre del programa a dar de alta.
- Solicita la clave del programa en el sistema de tráfico. Dentro del sistema de tráfico se debe dar de alta previamente la producción antes de utilizar esta aplicación, y el sistema de tráfico le entrega una clave al usuario. En nuestro módulo se debe anotar la clave para dar de alta el programa, la clave proporcionada es validada dentro del catálogo de producciones del sistema de tráfico a través de un DBLink, se obtiene el identificador del programa en el sistema de tráfico para ser guardado en nuestra base de datos con el objetivo de posteriormente calcular el código de inventario correspondiente a cada capítulo de la producción.
- El módulo informa acerca del éxito o si ha ocurrido algún problema en el proceso de alta. Para realizar el alta de la producción el módulo verifica que no se tenga un programa en la base de datos con el mismo identificador obtenido desde el sistema de tráfico que está asociado a la clave. Otra validación incluye que el nombre de la producción no esté ya dado de alta para evitar ambigüedades, si ya hay un programa con ese nombre informa al usuario acerca del problema.

El tiempo de desarrollo de esta aplicación fue de 3 días hábiles incluyendo pruebas y validación de los resultados.

3.4.3 Módulo interfaz WEB de alta de capítulos

El módulo de alta de capítulos es una aplicación que fue programada con lenguaje PHP del lado del servidor ejecutando en un servidor Apache bajo un sistema operativo Windows 7. Del lado del cliente también se ocupó HTML5, CSS3 y JavaScript para diferentes validaciones de los datos ingresados. La página WEB permitía registrar capítulos para una producción televisiva determinada.

En este módulo participé en el diseño de su funcionalidad así como la supervisión de su desarrollo e implementación, la codificación fue delegada a otro desarrollador del equipo de desarrollo a mi cargo.

El funcionamiento definido fue el siguiente:

- Muestra la lista de canales registrados en el sistema de automatización, los cuales están almacenados en la tabla de canales.
- Una vez elegido el canal muestra la lista de producciones televisivas que tiene registradas. El usuario debe seleccionar alguna producción para dar de alta un nuevo capítulo.

- El formulario incluye un campo para introducir el número de capítulo, el cual se valida que sea numérico y de hasta 4 posiciones.
- El formulario muestra un calendario para asignar la fecha de estreno planeada para el capítulo. La fecha de estreno planeada es una guía para que los usuarios programadores de guías que registraban los capítulos pudieran llevar a cabo un seguimiento adecuado. Los programadores de las guías también eran los encargados de elaborar las guías de programación dentro del sistema de tráfico por lo cual conocen estas fechas. El registro de la fecha planeada de estreno también era útil para que los editores conocieran cuánto tiempo queda para la emisión del programa y tuvieran organizados los materiales.
- El módulo cuenta un botón para dar de alta el capítulo, al ser pulsado se validan los datos introducidos con JavaScript y del lado del servidor se valida que el número de capítulo no exista dentro de la tabla de capítulos para la producción elegida. Si no existe lo agrega a la tabla de capítulos, asigna su identificador e inicializa los campos de la tabla a los valores correspondientes al estado inicial del registro. Esta operación de alta también calcula el número de sub código de inventario del capítulo, el cual en este caso no incluye los dígitos identificadores de bloques sino que en este punto es de una longitud de 14 caracteres, los dos dígitos que faltan correspondientes a los bloques se crean en otro punto del sistema. En caso de no superar alguna validación la página informa el usuario acerca del error.

El tiempo de desarrollo de esta aplicación también fue de 3 días incluyendo las pruebas y validación de los resultados así como ajustes en la interfaz. El diseño de las interfaces también lo supervisé para garantizar que fueran funcionales y fáciles de utilizar así como guardar ciertas proporciones estéticas, alineación y ortografía antes de poder ser aceptadas como candidatos para su liberación.

3.4.4 Módulo interfaz WEB ingestión de programas

La interfaz WEB de ingestión de programas se desarrolló en lenguaje de programación PHP del lado del servidor, se ejecutaba utilizando un servidor Apache bajo sistema operativo Windows 7. En la aplicación del lado del cliente se utilizó JavaScript, CSS3 y HTML5. Participé llevando a cabo su diseño y la supervisión de la implementación, la codificación la llevó a cabo otro miembro de mi equipo de desarrollo.

Esta aplicación web consta de dos partes, en la primera se tiene una interfaz gráfica que muestra la lista de capítulos registrados para la fecha de estreno elegida por el usuario, su funcionamiento es el siguiente:

- El usuario selecciona de una lista el canal con el cual requiere trabajar.
- Una vez elegido el canal, la aplicación despliega por default la lista de los programas registrados a estrenarse en la fecha actual.

- Si el usuario necesita revisar los registros de otra fecha incluye un calendario donde puede elegir la fecha adecuada y un botón de actualizar para refrescar los resultados.
- La interfaz muestra una tabla con los resultados la cual tiene como encabezado la fecha consultada. La tabla contiene un renglón por cada capítulo registrado para estreno en la fecha seleccionada. Los datos desplegados muestran el número de capítulo, el título y el estado tanto del proceso de ingestión como del proceso de revisión automática, que es una parte del sistema encargada de enviar el material ingestado a los servidores de almacenamiento. Los estados de la ingestión pueden ser nuevo cuando aún no se ha enviado un material, procesando cuando el material ha sido enviado al servidor y está siendo transcodificado, terminado si el proceso de transcodificación concluyó y puede ser rechazado en caso de que el sistema detecte un problema en alguna parte del proceso de ingestión o transcodificación. Los estados de la revisión automática solamente incluyen dos, nuevo si aún no se ha realizado o revisado si el resultado del proceso de ingestión es válido para enviarse a los servidores de almacenamiento.
- El usuario puede seleccionar alguno de los registros, si el estado de la ingestión es nuevo al darle clic continúa a la segunda parte del proceso de ingestión. Si el estado es procesando el usuario debe esperar y no realiza acción alguna la interfaz. Una vez que el proceso de ingestión ha finalizado y el registro cuenta con el estado terminado, al darle clic al registro se muestra un previo del video que ha sido ingestado anteriormente en baja resolución así como información adicional que incluye la duración de los bloques, duración total del capítulo, resolución y tamaño en disco de la alta resolución.

La segunda parte del módulo de interfaz de ingestión se activa al seleccionar un registro con estado de ingestión nuevo, lo cual envía al usuario a una interfaz secundaria en la que se permite al usuario seleccionar un archivo de su equipo y calificar los tiempos de inicio y fin de cada bloque con precisión de cuadros. Se permite capturar hasta un máximo de 10 bloques con sus respectivos tiempos de entrada y salida.

La televisión mexicana que se rige por los estándares NTSC o ATCS, en los cuales el framerate profesional es de 29.97 cuadros, para realizar la transmisión a esa tasa exacta hay diferentes mecanismos pero comúnmente se descartan algunas estampas de tiempo durante la reproducción. Esta cifra en sí es una característica de retro compatibilidad que se ha heredado a través del tiempo desde los sistemas de transmisión análogos y se generó debido a algunas pérdidas de precisión en equipos de esa naturaleza.

Una vez que se han calificado los puntos de inicio y fin de todos los bloques que el usuario requiere, la aplicación valida desde el lado del cliente que los puntos de inicio y fin sean congruentes así como que no exista traslape entre los bloques. La naturaleza del correcto formato en el texto de los timecode también es validada en este punto. Si las validaciones se superan se muestra un mensaje al usuario con un resumen de la información capturada para que confirme que los datos son correctos. Al tener la confirmación por parte del usuario la aplicación envía el archivo y los datos mediante una

petición HTTP al servidor, colocándolo en un disco destinado para el almacenamiento temporal del material. El nombre del archivo final en el servidor corresponde al ID del registro seleccionado en la tabla de capítulos para facilitar su posterior procesamiento.

La aplicación además lleva un registro de las acciones más importantes que va realizando. El manejo de bitácoras es un punto muy importante en este tipo de sistemas automatizados ya que permite en el caso de falla realizar un rastreo de la misma así como el diagnóstico y erradicación del problema, de hecho todas las aplicaciones desarrolladas en el área deben manejar una bitácora.

Si el proceso de envío es exitoso y el archivo es depositado en la ubicación correcta se marca en la base de datos que ha llegado al servidor, así como un mensaje de éxito o error para informar al usuario que realiza la carga el resultado de la operación. El archivo enviado queda a la espera del siguiente proceso automático: la transcodificación.

El servidor donde se reciben estos materiales contaba con 2 tarjetas de red, una de ellas se dedicó al envío y recepción de materiales hacia otros equipos mediante la interfaces web, y la otra como detallaré más adelante se reservó para el proceso de transcodificación. Debido a la demanda de poder de cómputo para recomprimir el material el proceso de transcodificación se realiza con una granja o conjunto de equipos.

Para el desarrollo de este módulo se dedicó 1 semana y un periodo de pruebas de otra semana más en el cual se validó el correcto funcionamiento de las transferencias de los archivos desde las cabinas de edición hasta el servidor, también se incluyeron pruebas con diferentes tipos materiales y se validaron que los tiempos de transferencia fueran los adecuados para una operación real.

El módulo tiene como restricción el envío de un archivo de video a la vez al servidor de ingestión. Como trabajo futuro se ha contemplado implementar el envío de múltiples videos al mismo tiempo o bien manejar una cola de envíos. El proceso de envío típicamente para archivos de duración de una hora toma de 8 a 10 minutos, aunque depende del tamaño del archivo y por lo tanto de las características internas del archivo, la duración del archivo y el tráfico de la red en el momento del envío. El envío múltiple es una característica que se está analizando porque dependiendo del mecanismo utilizado solamente se lograría dividir el ancho de banda de la red entre las transferencias simultáneas ocasionando que el proceso sea más lento comparado con la transferencia de un material a la vez.

3.4.5 Módulo transcodificación de programas

Los archivos de video que eran enviados desde las cabinas de edición mediante la interfaz WEB de ingestión podían tener diferentes características dependiendo de la configuración del programa de edición en el cual eran procesados, incluso podrían tener diferentes características aún cuando provengan del mismo equipo ya que dependen en gran medida de la metodología de trabajo del editor.

Existían producciones en resolución HD, en el caso de la televisión bajo el estándar ATSC la resolución HD para transmisión profesional es de 1920 pixeles de ancho por 1080 pixeles de alto, aunque también se manejaban producciones aún en resolución SD, principalmente aquéllas que fueron producidas antes de la adopción de los formatos HD pero aún continuaban siendo parte de la programación, esta resolución del estándar del formato HD se describe en Advanced Television Systems Committee (2009). La resolución SD para la televisión bajo el estándar NTSC es de 720 pixeles de ancho por 480 pixeles de alto. Un requerimiento fue que la resolución original de la imagen se debería conservar en los archivos ingestados. En cuanto al audio todas las producciones utilizadas contenían 2 canales de audio, esta característica también debería ser conservada en los archivos almacenados en el subsistema de almacenamiento desde el cual los reproductores los descargarían para realizar la transmisión.

Sin embargo existen algunas otras características de los archivos digitales, como el bitrate y el codificador con el cual están comprimidos, que marcan diferencias significativas entre unos y otros. Hay una gran variedad de codificadores desarrollados para diferentes propósitos, algunos son con pérdidas otros sin pérdidas y cada uno tiene sus propias características internas así como algoritmos de compresión. Algunos tienen mayores tasas de compresión de imagen logrando reducir más enormemente la cantidad de información necesaria para almacenar el video sin tener una pérdida significativa de calidad. Por otro lado tenemos otros formatos que no comprimen tanto la información pero conservan casi intacta la calidad de la imagen, algunos otros requieren de una mayor capacidad de cómputo, entre otras características. Entre toda la variedad de codificadores tanto de audio como video siempre se deben analizar sus ventajas y desventajas para elegir uno que sea idóneo para las tareas en las cuales se va a ocupar.

La gran variedad de codificadores que existen se debe en parte a los desarrollos de distintas empresas dedicadas al desarrollo de tecnologías de compresión, también existen asociaciones de varias empresas que generan estándares y por otro lado los avances tecnológicos. Sin embargo se debe analizar el propósito para el cual han sido diseñados los codificadores. Algunos formatos son ideales para almacenar la información de video y utilizarla posteriormente en procesos de edición como es el caso del códec ProRes de Apple que se incluye en el popular software de edición de video Final Cut, de hecho este software es muy utilizado en la industria de la televisión, generalmente maneja un bitrate muy alto para no perder información y guardar el video en formato digital con una alta calidad permitiendo utilizarlo en las ediciones posteriores para crear spots o programas completos, está optimizado para cargar rápidamente archivos muy grandes.

Existen otros compresores como el MPEG-2 que es un estándar para la codificación de audio y video diseñado por un grupo de alrededor de 20 corporaciones e incluso cuenta con la participación de la Universidad de Columbia. Este estándar es de muy amplio uso actualmente en la industria de la televisión ya que es soportado por cámaras de video profesionales, VTR profesionales, sistemas de ingestión e incluso para la transmisión de señal al aire en televisión en los estándares ATSC y DVB. El uso para la transmisión al aire de MPEG2 en el estándar ATSC está descrito en Advanced Television Systems Committee (2009) y respectivamente para DVB en DVB (2018). Sin embargo para que los compresores implementados bajo este estándar mantengan una buena calidad de imagen también requiere el manejo de un bitrate relativamente alto, de lo contrario se observarán macrobloques que el usuario final percibe como cuadrículas notorias en la imagen principalmente en escenas con movimientos rápidos debido a los algoritmos que utiliza al realizar la compresión del video.

El estándar MPEG-4 es el sucesor de los estándares MPEG-2, también fue creado por un grupo de corporaciones y expertos, sin embargo debido a que los algoritmos son más avanzados permite conservar calidad de imagen igual o superior a un codificador MPEG-2 con bitrate igual o incluso menor. Este formato está siendo muy utilizado en la industria para almacenar los contenidos de video en archivos de tamaño mucho menor al de los formatos utilizados para edición y la calidad que tienen es aceptada para realizar la reproducción de señales tanto en resolución HD como SD. El estándar MPEG-4 define una rama llamada MPEG-4 parte 10 que también recibe el nombre de H264 o AVC (Advanced Video Coding) y es el estándar teórico utilizado por los códecs disponibles en el mercad, como lo detalla The Moving Picture Experts Group (2002). En este estándar el audio utiliza generalmente la compresión AAC (Advanced Audio Coding) aunque en realidad proviene desde la definición del estándar MPEG-2 su rendimiento y calidad hacen que también sean parte de MPEG-4.

Debido a la eficacia que tienen los codificadores desarrollados bajo el estándar MPEG-4, su uso probado en sistemas de almacenamiento, el soporte que tienen los navegadores WEB con HTML5 para su reproducción y que nos facilitaría el desarrollo de la visualización de los previos fue elegido como el formato para almacenamiento de los videos tanto en alta resolución como en baja resolución.

En el momento del desarrollo del sistema de automatización de máster eran pocos los sistemas comerciales que ofrecían soporte de almacenamiento de video en formato MPEG-4 por lo que también fue una elección para adoptar tecnología de la última generación disponible.

El procesar todos los videos para normalizar sus características, aunque requiere recursos de cómputo y tiempo, es ideal para que todos los archivos se encuentren en el mismo formato y de esta forma todos los procesos del sistema incluyendo el reproductor final tengan un desempeño similar al leer cualquier archivo ingestado.

Otro de los objetivos de la normalización es optimizar el espacio de almacenamiento. Se habían tenido experiencias en otros sistemas donde los usuarios pueden ingresar

archivos a bitrate muy alto cuando no vale la pena respecto a la calidad o resolución del archivo, también esas prácticas pueden llevar a tener archivos almacenados incluso de duraciones pequeñas pero que ocupan mucho espacio en disco. En la investigación de las prácticas que llevan a cabo otras soluciones se observó que varias proponen el mismo tratamiento debido a las razones antes expuestas.

Como señala Sencore (2011) el proceso mediante el cual se cambia el códec de un archivo digital a otro tipo de compresión utilizando otro codificador es llamado transcodificación. Este proceso generalmente requiere descomprimir el contenido del archivo para luego comprimirlo con el nuevo codificador.

El módulo de transcodificación fue diseñado para procesar cada archivo correspondiente a un capítulo de una producción que se envíe desde las cabinas de edición al servidor web, generando como resultado un archivo igual al enviado en baja resolución y una serie de archivos en alta resolución dividiendo cada uno de los bloques del capítulo. En ambos casos el video es codificado en H264 y al audio en AAC, ambos son encapsulados en un contenedor mp4.

Antes de iniciar el procesamiento de un archivo, el módulo de transcodificación realiza consultas a la base de datos para determinar si el nombre del archivo coincide con el identificador de algún capítulo en las tablas de ingestión, si hay coincidencia se obtiene el estado del registro del capítulo así como otros datos almacenados. Con los datos obtenidos realiza algunas verificaciones que incluyen las siguientes:

- Si el nombre del archivo no coincide con ningún identificador de algún registro de capítulos el material es rechazado.
- Si el estado de transcodificación no corresponde a un estado inicial, el archivo es rechazado.
- Valida los tiempos de inicio y fin de cada bloque introducidos por el usuario, verificando que los tiempos de inicio sean menores a los de fin de cada bloque, también que los tiempos de inicio y fin de los bloques no se traslapen y todos los bloques deben estar contenidos dentro de la duración del archivo a procesar.
- Verifica las propiedades de archivo entre las cuales está la resolución de los videos a 1920 por 1080 en el caso de HD y 720 por 480 en el caso de SD. Estas son las únicas resoluciones válidas porque el aplicar una transformación podría deformar la imagen, por lo anterior se acordó que los editores deben entregar el material en la resolución adecuada.
- El archivo a transcodificar debe tener 2 canales de audio.
- El framerate del video debe ser 29.97 cuadros por segundo, que es el estándar para la televisión.

En caso de fallar en alguna de las verificaciones se actualiza el registro del capítulo en la base de datos guardando en el campo de estado de ingestión un valor que significa rechazado, además el video es enviado a una carpeta de materiales rechazados para su respaldo y en caso de ser necesario su posterior diagnóstico con evidencia.

El proceso de transcodificación genera 2 tipos de archivos de salida, el primer tipo es un archivo por cada bloque que envió el usuario dividiendo el video original en los tiempos de inicio y fin de cada bloque. Al final del proceso se tiene como salida un archivo de video por bloque en alta resolución, es decir, en resolución HD o SD con las dimensiones originales antes descritas. El bitrate de los archivos de salida está optimizado para conservar una buena calidad de imagen siendo 12 Mbps en HD y 8 Mbps en SD.

El segundo tipo de archivo es una copia del archivo ingestado transcodificado en baja resolución, la cual es de un cuarto de pantalla del original. Los archivos de baja resolución tienen por objetivo almacenarlos para la vista previa en las interfaces web del módulo de ingestión. En este caso el bitrate para los videos cuyo material original es HD es de 4 Mbps y para los SD 1.2 Mbps, los cuales son más bajos que los utilizados para almacenar los videos destinados a la transmisión. El bitrate es significativamente menor porque estos videos solamente se utilizan para mostrar una vista previa y tener una muestra del archivo original tal como se envió. La información para visualizar el previo viaja por la red por lo que es necesario no saturarla con contenidos de alta resolución y bitrate elevado, además el servidor dedicado para almacenar la baja resolución contaba con menos espacio que el dedicado para el almacenamiento de los videos en alta resolución.

La generación de ambos tipos de archivo es un proceso que se realiza de manera simultánea con la finalidad de realizarlo en el menor tiempo posible y también aprovechar en el mayor nivel posible los recursos del procesador del equipo donde se realiza la tarea.

El proceso en SD se realiza a una velocidad alrededor de 4x, es decir, dependiendo de la duración real del video le toma alrededor de un cuarto de esa duración procesar y generar ambos archivos. Para los archivos en resolución HD la velocidad del proceso es de 2x, es decir, la mitad del tiempo real que dura el archivo a procesar.

Aunque la interfaz de envío WEB solo permitía enviar un archivo a la vez, se tenían varias cabinas de edición y el proceso de transcodificación era más tardado que el de envío, si se tuviera solamente un transcodificador podrían acumularse los archivos en espera de ser procesados llevando al sistema a una condición de sobreflujo. Por lo anterior se destinaron varios equipos para realizar la tarea de transcodificación y procesar más de un archivo a la vez. Cada proceso de transcodificación ocupa el procesador del equipo en alrededor del 75% al 80% de sus recursos, por lo que se diseñó para ejecutar un proceso por equipo. Se instalaron 6 equipos para atender la demanda de transcodificación de archivos enviados desde las cabinas de edición y con la proyección para atender hasta 5 canales con todas sus producciones con el sistema de automatización desarrollado.

Este conjunto de equipos constituye lo que generalmente en estos sistemas se le denomina granja de transcodificación o bien granja de procesamiento. La unidad donde se reciben los archivos en el servidor de ingestión se configuró como una carpeta compartida y esta se mapeó en los equipos transcodificadores. Los equipos transcodificadores leen el archivo original mediante la unidad compartida, a través de la red, y generan los archivos de salida en sus discos duros locales.

Una vez que el proceso ha terminado, y si este ha resultado exitoso, los archivos de salida se mueven a carpetas compartidas de regreso al servidor de ingestión, hay una destinada para los archivos de baja resolución y otra para los de alta resolución. El proceso de transcodificación es totalmente automático y si detecta un problema guarda el estado de error en la base de datos y el archivo es depositado en una carpeta compartida desde el servidor de ingestión pero que solamente recibe los videos con defectos o resultados del procesamiento incorrecto, esto con fines de diagnóstico.

Cada transcodificador tiene un identificador asignado el cual es guardado en el registro del capítulo al que corresponde el archivo en la base de datos con el objetivo de conocer el equipo que se encargó del procesamiento en caso de cualquier fallo, también registra el número de bloques que fueron cortados en el proceso. Los nombres de los archivos de cada uno de los bloques en alta resolución contienen el identificador del capítulo en el sistema de ingestión más un guión bajo y el número de bloque correspondiente. Al terminar el proceso de manera exitosa los archivos originales son eliminados.

Participé en el diseño, desarrollo y programación del módulo de transcodificación. La aplicación fue programada utilizando el lenguaje C++ y se apoya en la integración de la librería MediaInfo y librerías basadas en libav. La librería MediaInfo se utilizó para obtener las propiedades de los archivos tales como la duración, resolución, número de canales de audio, etc. Por otro lado libav es un conjunto de librerías de código abierto dedicadas a la reproducción, decodificación y codificación de archivos de audio y video digitales. La lista de formatos y códecs que puede manejar es muy amplia tanto en audio como en video, incluyendo todos los formatos utilizados en la industria de la televisión y los que eran utilizados en el software de edición de las cabinas de producción. La biblioteca libav es parte del reproductor de código abierto VLC. Las aplicaciones que se crearon en C++ además ocuparon parte del framework que se ha generado en el área en cuyo desarrollo también he participado para realizar tareas básicas incluyendo clases para la generación de bitácoras en archivos de texto plano. La aplicación genera un archivo de bitácora por día. Otra de las clases que siempre se deben incluir son las dedicadas a crear archivos de configuración mediante los cuales es posible definir parámetros de operación como lo son el ID del transcodificador, el bitrate de los archivos de alta y baja resolución así como otros detalles más. La utilización de archivos de configuración es muy común en todo tipo de software y da la ventaja de poder cambiar el comportamiento de las aplicaciones mediante la modificación de parámetros sin la necesidad de volver a compilar los programas cada que se requiere un cambio.

Los transcodificadores fueron instalados sistemas operativos Windows 7. Integran además conexiones a base de datos Oracle para consultar los datos del capítulo así como actualizar sus estados, la clase de conexión de la base de datos también es parte del framework desarrollado en el área aunque integra las librerías dinámicas del cliente de Oracle, puede utilizar una versión del cliente 9i o superior.

El tiempo de desarrollo de esta aplicación fue de dos semanas incluyendo las pruebas con muestras de archivos como los enviados desde las cabinas de edición. Se realizaron pruebas con varios esquemas de configuración y se midió el rendimiento del uso de la

red, así como pruebas de estrés al sistema transcodiando en todos los equipos posibles simultáneamente. En las pruebas de estrés se determinó que era necesario dedicar la segunda tarjeta del servidor de ingestión para la conexión a los transcodificadores lo cual se logró modificando las tablas de ruteo. Tanto el servidor de ingestión como los transcodificadores se conectaron al mismo switch de datos y se configuraron en una red dedicada para los módulos del subsistema de ingestión.

Como área de oportunidad se planea en el futuro automatizar aún más el proceso de ingestión con la inserción de algún tipo de imagen especial que represente el inicio de un bloque y otra diferente para el fin de bloque. Las imágenes especiales podrían ser insertadas dentro del video durante las tareas de edición, y los transcodificadores entonces tendrían que detectar estos cuadros especiales para realizar los cortes en automático. De esta forma el usuario ya no tendría la necesidad de capturar los tiempos de inicio y fin de cada bloque, lo cual podría evitar errores en la captura de los tiempos.

3.4.6 Almacenamiento de los programas en baja resolución

En el proceso de transcodificación se genera un archivo de baja resolución por cada archivo enviado desde las cabinas de edición, este archivo es una copia del archivo original transcodificado a formato MPEG-4 con el objetivo de mostrar una vista del resultado de la transcodificación. Como ya se mencionó este archivo se deposita en una carpeta compartida destinada para el almacenamiento de estos archivos en baja resolución en el servidor de ingestión. El nombre de los archivos de baja resolución es el identificador del capítulo en la base de datos del sistema de ingestión. En el servidor de ingestión se dedicaron varios discos para el almacenamiento estos archivos de baja resolución.

En la tabla que guarda el registro de los capítulos registrados para cada producción en el subsistema de ingestión tiene un campo que indica el disco en donde está almacenado el archivo en baja resolución.

Los transcodificadores fueron configurados para entregar siempre en la misma carpeta los archivos de baja resolución, sin embargo con el paso del tiempo el disco de almacenamiento se va llenando. Por lo anterior se ha definido el campo que indica el disco donde está almacenado el archivo de baja resolución del capítulo con el objetivo de soportar varios discos de almacenamiento. Se desarrolló además una aplicación sencilla programada en lenguaje C++ que se encarga de mover el material más viejo a otro disco cuando el disco donde los transcodificadores depositan los archivos de baja resolución alcanza un porcentaje ocupación del 80%, al mover el archivo actualiza en la base de datos la ruta del disco en donde es depositado.

El sistema guarda un catálogo con los discos duros posibles para almacenar los archivos de baja resolución en el servidor de ingestión, esto se hace a través de la base de datos donde se creó la tabla TVP_ALMAC_WEB para este fin. En esta tabla se tiene un campo

donde se guarda una cadena que representa el alias para cada uno de los discos duros, un identificador y la ruta física en el servidor web. Cada uno de los alias en el catálogo de los espacios de almacenamiento se configura en el servidor WEB Apache para que reconozca los discos mediante dicho alias y así se pueda tener acceso a los archivos de baja resolución mediante el servidor WEB. Por seguridad el servidor está configurado para impedir el listado de archivos en estos directorios así como la escritura.

En el servidor fueron configurados 3 discos duros destinados para el almacenamiento de los videos en baja resolución de 3TB cada uno, los cuales pueden contener en conjunto hasta 5 años de archivos de baja resolución para la visualización de previos.

3.4.7 Interfaz WEB para consulta de programas ingestados

La interfaz Web para la consulta de programas ingestados fue diseñada como una aplicación web con una interfaz gráfica similar a la de interfaz de envío de archivos de programas, salvo dos diferencias. La primera es que no se pueden enviar materiales desde esta aplicación y la segunda diferencia es que muestra las duraciones de cada bloque de forma detallada que resultaron del proceso de transcodificación para los usuarios que necesitan revisar los materiales ingestados y transcodificados exitosamente.

Ambas aplicaciones WEB se crearon con la capacidad de mostrar los archivos de baja resolución a los cuales tienen acceso mediante el servidor WEB Apache. Para realizar la visualización del video durante la carga de los registros se realiza una consulta a la base de datos en la cual se hace el cruce del campo que indica el disco en el cual está almacenado el video correspondiente de baja resolución y el catálogo con el alias de las rutas configuradas en el servidor WEB.

Con la información del alias WEB que le corresponde al registro, el código escrito en PHP determina la ruta completa en servidor para incluirla como parámetro de la fuente en un tag de video en HTML5 y así mostrar el video en la interfaz.

El soporte del tag de video HTML5 es una de las razones por las cuales el uso de esta aplicación WEB se recomendó en el navegador Google Chrome, ya que el soporte era completo en dicho navegador. El soporte para el tag proveía de todos los controles necesarios para la reproducción del video a mostrar como pausa, iniciar reproducción, barra de avance e incluso controles para llevarlo a pantalla completa.

Participé en el análisis de requerimientos y diseño de la funcionalidad de esta aplicación, la programación fue realizada por un programador de mi equipo de trabajo en 3 días incluyendo las pruebas. La aplicación fue programada en lenguaje PHP y se instaló en el mismo servidor web de aplicaciones que los otros módulos descritos anteriormente.

Con la exposición de esta aplicación web termina la presentación de los módulos dedicados a la ingestión de los programas, el proceso ha demostrado ser confiable al estar en operación a lo largo de 3 años sin ningún incidente significativo, solamente se han detectado videos rechazados por errores en la captura de datos de los tiempos de inicio y fin de los bloques o bien por errores en la validación de los parámetros básicos, en

cuyo caso los editores deben reenviar el material. Los módulos se diseñaron para funcionar de manera autónoma las 24 horas de día ya que las cabinas de edición podían entregar videos incluso durante el transcurso de la madrugada. La implementación del subsistema de ingestión dedicada a la recepción de los programas incluso permite entregar programas el mismo día en que se va a transmitir un capítulo por primera vez, para este fin se definió una regla según la cual el archivo de video con el capítulo completo debe ser enviado con al menos una hora de anticipación.

3.4.8 Módulo interfaz WEB ingestión de spots y bloques de programa

Uno de los requerimientos a cubrir fue el ingreso de spots al sistema. Los spots son materiales de video que contienen comerciales, promocionales de otras producciones o cualquier otro tipo de campañas publicitarias. Para realizar este proceso fue necesario el trabajo colaborativo con el sistema de tráfico.

En el sistema de tráfico los usuarios programadores de las guías dan de alta los registros que contienen la información de cada spot. El sistema de tráfico también realiza el alta del registro del spot con parte de la información ingresada en una tabla en la base de datos del sistema de automatización a través de un DBLink. La tabla para el registro de los materiales se nombró TVP_MAT_TRANS, su estructura se revisará más adelante en el subsistema de almacenamiento.

Cada uno de los spots tiene un registro en esta tabla, este fue un proceso derivado de una interfaz ya existente entre el sistema de tráfico y otros proyectos desarrollados por el Departamento de Desarrollo Tecnológico, dicha interfaz se adaptó para minimizar el tiempo de desarrollo y contar con una base de procesos sólidos ya probados.

Los spots cuentan con un identificador único y común entre ambos sistemas llamado código de inventario, el cual es numérico de 8 dígitos. Otros datos en cada registro son el nombre del cliente, el título de la versión y la duración planeada, entre otros.

La aplicación WEB para el envío de spots se creó para permitir al usuario seleccionar un archivo de su equipo y enviarlo al servidor de ingestión de spots. El archivo conservará su contenido y duración sin ningún tipo de corte o edición durante el proceso, de manera que se respeta íntegramente su contenido para su transmisión en la señal de salida del sistema de automatización.

El nombre del archivo seleccionado debe corresponder al código de inventario del spot para que este sea reconocido por el sistema y pueda relacionarse con el registro dado de alta en la tabla TVP_MAT_TRANS, la cual como se expone más adelante se utilizó en el reproductor que ejecuta la guía de programación para cargar los datos de los videos a reproducir. El servidor de ingestión de spots, es un equipo distinto al dedicado a la ingestión de programas, y una vez que el archivo es enviado queda en espera de ser validado para enviarlo al proceso de transcodificación correspondiente. Esta aplicación también tuvo la limitante de enviar un archivo a la vez.

Otro de los requerimientos planteados por los usuarios fue la posibilidad de enviar videos correspondientes a bloques de individuales de programas, porque en la operación diaria existían producciones cuyos contenidos no pueden ser entregados con el margen requerido por el proceso de entrega de programas completos, incluso algunos bloques correspondientes a un capítulo podían estar listos en el área de edición aproximadamente 15 minutos antes de que sea necesario transmitirlos al aire como lo son las producciones de noticias. Para cubrir esta operación el módulo le permite al usuario enviar un archivo con el contenido del bloque, el cual es depositado en el servidor de ingestión de spots a la espera de un proceso manual que permite atender ágilmente la ingesta al sistema de este tipo de materiales. Para el envío de bloques de programa individuales el nombre del archivo tiene mayor libertad exceptuando caracteres especiales los cuales se validaron en el formulario. El título del archivo puede llevar una descripción breve de su contenido. Los videos enviados de esta forma deben contener solamente el bloque de programa a transmitir ya que se importará íntegramente al sistema para su reproducción.

A los archivos que son enviados al servidor mediante esta aplicación en el momento de ser almacenados en el servidor se les cambia el nombre por uno que resulta de la concatenación del nombre original dado por el usuario con la hora y fecha del envío para identificarlos plenamente, el mecanismo anterior también previene que los archivos no se sobrescriban si se envía otra versión del mismo bloque incluso con el mismo nombre.

La aplicación WEB para enviar spots y bloques de programa fue desarrollada en lenguaje PHP, se instaló en un servidor Apache ejecutando sobre un sistema operativo Windows 7. Para el lado del cliente se utilizó HTML, JavaScript y técnicas de AJAX. El tiempo de desarrollo llevó alrededor de 3 días con las pruebas necesarias. Participé en el diseño del módulo y la codificación fue desarrollada por otro programador del equipo de trabajo.

3.4.9 Módulo transcodificación automática de spots

Una vez que el archivo de video que contiene el spot o bien el bloque de programa es depositado en el servidor de ingestión de spots, se creó un módulo de transcodificación automática de spots que se encarga de leer los archivos desde la ubicación donde fueron depositados para transcodificar solamente aquellos que sean spots.

El módulo se diseñó para llevar a cabo una validación y determinar si se trata de un spot o un bloque de programa, la cual se realiza mediante consultas a la base de datos. Antes de realizar la consulta verifica si la longitud del nombre del archivo original es de 8 dígitos y que el nombre sea numérico, si es estas condiciones se cumplen entonces busca en la tabla TVP_MAT_TRANS de la base de datos el registro que tenga el código de inventario igual al número en el nombre del archivo original. El módulo debe descartar del nombre del archivo la hora y fecha que se concatenaron al nombre original en el proceso de envío al servidor antes de obtener el código de inventario para realizar la validación.

La consulta realizada a la base de datos obtiene varios campos del registro del spot, entre los más importantes determina si el spot ya tiene un archivo de video asignado previamente y la duración planeada para el spot cuando el programador de las guías lo dio de alta en el sistema de tráfico. Con los datos obtenidos el módulo que realiza el proceso automático es capaz de tomar decisiones, si el registro ya cuenta con un video asignado previamente entonces el archivo a procesar se descarta siendo respaldado en una carpeta que almacena todos los videos con materiales ya ingestados con el objetivo de identificar los que han sido descartados por este motivo. Los vídeos rechazados son catalogados en subcarpetas por fecha. La duración también es validada, el video a procesar debe tener la duración indicada en el registro con un margen de error de máximo medio segundo, si la duración del archivo no es la adecuada entonces el archivo es descartado moviéndolo a una carpeta dedicada a almacenar los archivos rechazados por duración errónea, en este caso también la aplicación organiza los materiales en subcarpetas que tienen por nombre la fecha del envío para tener los archivos catalogados. Los parámetros básicos del archivo también se validan de una forma semejante a los bloques de programas entre los cuales están la resolución, el framerate y el número de canales de audio, si no cumple con ellos también hay una carpeta donde se respaldan los videos con parámetros erróneos y son catalogados por la fecha de envío.

El módulo de transcodificación realiza una revisión de la carpeta donde el módulo de ingestión web de spots y bloques de programa deposita los archivos enviados para buscar si existen nuevos videos cada 5 minutos.

El archivo que ha pasado todas las verificaciones se mueve a una carpeta temporal para su procesamiento desapareciendo del directorio original donde fue depositado por el módulo de envío. El módulo entonces transcodifica el spot y dependiendo de la resolución del video elige el bitrate adecuado siendo el mismo que en el módulo de transcodificación de programas completos, dependiendo de la resolución.

El programa es capaz de verificar el resultado del proceso mediante otra serie de validaciones que incluyen los parámetros del archivo resultante y la duración del archivo de salida que debe ser la misma que la del archivo original. Los archivos resultantes del proceso exitoso se guardan en una carpeta en la cual quedan a la espera del módulo que se encarga de enviar los spots a los servidores de almacenamiento.

El usuario recibe vía correo electrónico una notificación en la que especifica el resultado del proceso de transcodificación de cada spot enviado al sistema, con la cual conoce si el proceso fue exitoso o no. Los correos electrónicos se envían mediante un servidor de correo que se conecta a los servidores de correo electrónico disponibles en la red donde se instala utilizando el protocolo SMTP. Participé en el diseño y programación del servidor de correo electrónico, el cual es un componente que se utiliza en una gran cantidad de aplicaciones desarrolladas en el área para realizar esta tarea.

El módulo de transcodificación automática de spots fue desarrollado en lenguaje C++ con la integración de la librería MediaInfo y librerías basadas en la biblioteca libav que permite realizar el proceso de transcodificación. Participé en el diseño, codificación y desarrollo de este módulo en un tiempo de una semana incluyendo las pruebas.

El módulo de transcodificación automática de spots ha demostrado ser altamente confiable ya que se encuentra en operación las 24 hrs del día todo el año procesando los spots que son enviados desde edición, únicamente se ha interrumpido el servicio de forma programada durante los 4 años que lleva en operación para mantenimientos del servidor, la red o de la base de datos. Como trabajo futuro y área de oportunidad se encuentra el generar la baja resolución de los spots para tener una vista previa como en el proceso de ingestión de programas completos.

3.4.10 Módulo catálogo de spots ingestados

El catálogo de spots ingestados es una aplicación WEB que permite visualizar el estado de los spots dados de alta en la base de datos indicando en una tabla los datos de cada spot entre los cuales se encuentra el nombre, duración, código de inventario, código de material y el estado del material que muestra si el spot ya ha sido entregado al sistema, transcodificado y está listo para la transmisión.

La interfaz del módulo contiene controles para buscar spots en la base de datos por código de inventario o bien por el nombre de la versión con el objetivo de filtrar el listado de spots que se muestran.

La aplicación WEB fue diseñada para ser utilizada por todos los usuarios que participan en los procesos de planeación de guías, edición, supervisión de entrega de materiales y ejecutivos de venta, de esta forma contarían con una herramienta para estar enterados del estado de los spots sin la necesidad de recurrir solamente al envío de notificaciones.

El módulo fue desarrollado en lenguaje de programación PHP, se instaló en un servidor WEB Apache en un servidor con sistema operativo Windows 7. Participé en el diseño de este módulo y la supervisión de su implementación. El desarrollo fue terminado en 3 días incluyendo pruebas para asegurar su funcionamiento.

3.4.11 Módulo de Ingestión manual

El procesamiento de los bloques de programas individuales exige ser atendido en el menor tiempo posible, por otro lado cada bloque de programa debía ser revisado porque el material se transmitía al aire casi inmediatamente después de pasar por este proceso. Para cubrir estos requerimientos se definió la operación manual de este proceso, una vez que el bloque de programa es enviado al servidor web de ingestión se identifica que los nombres de estos archivos no contienen un código de inventario registrado en la base de datos y además se valida si la duración del material es superior a 5 minutos, materiales con esas características no son enviados al proceso de recodificación de spots ya que son bloques de programa. Estos materiales se quedan en el directorio de recepción del servidor, dicho directorio fue montado como una unidad compartida en un equipo al cual el usuario operador de máster tiene acceso. En este último equipo se instaló el módulo de ingestión manual, el cual es parte de un sistema más complejo de automatización de inserción de comerciales. Una actividad importante dentro de la ingeniería es la posibilidad de reutilizar componentes que permitan dar solución a las necesidades en el menor tiempo posible y de la mejor manera. Para cumplir con este objetivo se llevaron a cabo unas pocas adecuaciones en la aplicación para utilizarla dentro de este proceso.

Algunas de sus tareas son:

- Acceso a la base de datos del sistema de spots y programas permitiendo filtrar la información con base en ciertos criterios.
- Permite importar un archivo de audio y video en cualquier registro nuevo, es decir, que no tiene material asignado. Al realizar este proceso además transcodifica el material en MPEG-4 con parámetros similares al material ingestado mediante las interfaces WEB.
- La operación de importar materiales se puede hacer tanto en registros que son spots como en bloques de programas.
- Permite la creación del contenido tipo “súper”, los cuales son secuencias de imágenes o animaciones que se sobreponen a la señal ocupando una porción de la pantalla.
- Mediante este módulo es posible borrar el material del subsistema de almacenamiento de un registro seleccionado.
- Envío integrado utilizando el protocolo FTP al subsistema de almacenamiento y también directamente a los reproductores del sistema de automatización.

Cabe señalar que este módulo permite acceder al nivel de organización en la base de datos donde los registros representan spots individuales o bloques de programa

individuales, esta información se almacena en la tabla TVP_MAT_TRANS. Desde el sistema de tráfico se ingresan los registros de los bloques de programas con valores en los campos de manera que representan registros sin material en el momento en que el programador de la guía termina de crear la guía planeada del día siguiente.

Los módulos que se explicarán más adelante para el envío de material al subsistema de almacenamiento pueden convivir con las operaciones de ingestión manuales cuyos registros vacíos como se mencionó se crean a través del DBLink desde el sistema de tráfico. Así mismo el proceso dentro del sistema de tráfico fue diseñado para no dar de alta registros de materiales que ya tengan material asignado. Este diseño permitió el flujo de trabajo de los programas donde los materiales no se entregan con anticipación al sistema y puede convivir con los procesos descritos anteriormente donde el material si se entrega con días de anticipación.

El módulo de ingestión manual no fue ocupado en los procesos de ingestión de spots debido a que solamente está disponible para ejecutar en el sistema operativo Windows, por lo que el requerimiento de envío desde cabinas de edición en múltiples plataformas no se cumple.

Por otro lado este módulo manual tenía soporte para otras funciones tales como depurar algún registro de spot, esto es útil por ejemplo para realizar la corrección de un spot. Para corregir un spot se debe borrar desde esta aplicación el archivo de video del sistema y marcar el registro como vacío, posteriormente se puede volver a enviar el archivo con la corrección mediante las interfaces WEB con el flujo de trabajo habitual.

En cuanto a los sùpers, como se ha mencionado son imágenes superpuestas que pueden contener comerciales de anunciantes diversos sobre la imagen en pantalla como se pueden observar en las transmisiones de eventos deportivos, números telefónicos para la adquisición de algún producto u otra información. También este tipo de materiales pueden contener la clasificación del programa, información promocional del canal como horarios, indicaciones de retransmisión o un ejemplo claro es la leyenda “estás viendo” que muestra al televidente el nombre del programa actual. Estos materiales generalmente ocupan solamente una porción de la pantalla, tienen la posibilidad de tener animaciones y transparencias en las imágenes y carecen de audio.

Para la generación de los sùpers se definió que el usuario entregara al módulo de ingestión manual una secuencia de imágenes que pueden ser en formato PNG o TGA, el módulo se encarga de validar que la resolución de cada imagen tenga las dimensiones adecuadas para ser considerado en formato HD o SD. Además en cada imagen se valida que cumpla con los parámetros necesarios y contener un canal de transparencia o también llamado alpha channel. Dado que el framerate de la televisión es de 29.97 cuadros por segundo, dependiendo de la cantidad de imágenes entregadas será la duración del súper generado en el módulo. El conjunto de imágenes se encapsula en un contenedor MOV utilizando el API de QuickTime. El resultado es un solo archivo de video que contiene la secuencia de imágenes, el cual es relacionado con un registro y de esta manera también se asigna a un código de inventario con el cual puede ser identificado en

todas las operaciones del sistema. Los sùpers son enviados desde este módulo vía FTP a los reproductores finales donde se utilizará el material para su transmisión.

Realicé el diseño de este módulo así como su desarrollo e implementación en lenguaje de programación C++. También contó con la integración de las librerías basadas en Libav, la integración de cliente de base de datos Oracle y como ya se mencionó el API de QuickTime que fue creada por Apple para la manipulación de audio y video. En cuanto a las transferencias mediante FTP se realizó con una clase desarrollada utilizando la API WinINet que ofrece el sistema operativo Windows y esta clase se ocupó en todos los componentes del sistema que realizan este tipo de transferencias de archivo.

El módulo de ingestión manual fue uno de los primeros módulos que tuve a mi cargo desde el inicio de mi participación en el equipo de trabajo, cuenta con un desarrollo resultado de años de investigación, mejoras y la adición de nuevas características. En particular los ajustes necesarios para el sistema de automatización de máster tomaron una semana incluyendo las pruebas necesarias para validar su funcionamiento.

3.4.12 Herramientas auxiliares

Borrado de programas

En ocasiones era necesario borrar un programa que hubiera sido ingestado y transcodificado dentro del sistema, esto se podía deber a varias razones como lo son una corrección del contenido del programa, el programa pudo haber sido rechazado porque alguna de sus características no era la adecuada o los tiempos ingresados son erróneos bajo las validaciones descritas en la exposición del módulo de transcodificación, también puede ser que el editor hubiera enviado el archivo equivocado.

El proceso para borrar bloques de programa individuales o spots fue diseñado como ya se expuso para ser realizado con el módulo de ingestión manual.

Para borrar los programas completos que se hubieran entregado mediante la Interfaz Web de Ingestión de Programas se desarrolló una aplicación en la cual se puede visualizar la lista de programas que han sido ingestados así como su estado. En esa lista es posible elegir un programa en el cual el estado sea ingestado, transcodificado y haya sido enviado al sistema de almacenamiento. La aplicación genera una petición de borrado que se almacena en la base de datos en la tabla TVP_PETICION_REINGES_PROG. Después un proceso automático se encarga de buscar las peticiones de borrado que se han solicitado y aún están pendientes. Las peticiones son atendidas borrando tanto el material en alta resolución como el material de baja resolución, los registros que se hayan generado en la base de datos como parte del proceso de envío al subsistema de almacenamiento en la tabla TVP_MAT_TRANS también son borrados y regresando el registro del capítulo en las tablas propias del subsistema de ingestión al estado de nuevo. Al final del proceso marca las peticiones atendidas con el estado terminado. El almacenar las peticiones en la base de datos permite mantener un histórico de las peticiones de

borrado de programa que es útil para revisar los procesos del área de edición encargada de enviar los materiales.

La herramienta para la atención de las peticiones de borrado fue diseñada como un servicio para que en el futuro pudiera ser creada una interfaz WEB multiplataforma encargada de la alta de las peticiones de reingestión, porque la aplicación actual de alta de peticiones es una aplicación de escritorio que se ejecuta solamente en un equipo. El control por lo tanto es centralizado, fue instalado solamente en un equipo en el cual el responsable realiza las solicitudes, sin embargo, el proceso podría ser ejecutado por el usuario que lo necesite para lo cual se tiene planeado el desarrollo de la interfaz WEB.

Tanto el servicio que atiende las peticiones como la aplicación que permite dar de alta las solicitudes de borrado fueron programados en lenguaje C++ con la integración del cliente de base de datos Oracle. Estas aplicaciones se instalaron en un sistema operativo Windows7. Realicé el diseño, desarrollo e implementación de estas herramientas en 3 días incluyendo pruebas.

3.5 Subsistema de almacenamiento

En general los sistemas de automatización de máster cuentan con algún tipo de módulo de almacenamiento que realiza esencialmente dos funciones. La primera como su nombre lo indica es almacenar el material de audio y video independientemente del almacenamiento temporal que se pueda tener en los reproductores, generalmente los módulos de almacenamiento cuentan con mucho mayor espacio para este propósito a diferencia de los reproductores. La segunda tarea de la cual se ocupa el módulo de almacenamiento en este tipo de sistemas es la administración del material incluyendo el resguardo de metadatos de los archivos de media que se van a reproducir, comúnmente cuentan con algún tipo de base de datos que facilita a los usuarios la búsqueda de los contenidos a transmitir y permiten procesos como la automatización de la carga de guías, detección de cambios de materiales en los reproductores y calificación de material, entre otras funciones.

En el sistema de automatización de máster que desarrollé el subsistema de almacenamiento se compone de dos servidores de almacenamiento, en un esquema espejo de principal y respaldo. La base de datos de este subsistema de almacenamiento se alojó en un servidor Oracle. Para construir el subsistema de almacenamiento se desarrollaron una serie de herramientas y módulos que permiten realizar el envío de los materiales hacia los servidores de almacenamiento así como la depuración de los mismos garantizando que siempre haya espacio disponible.

El siguiente diagrama muestra los componentes del subsistema de almacenamiento:

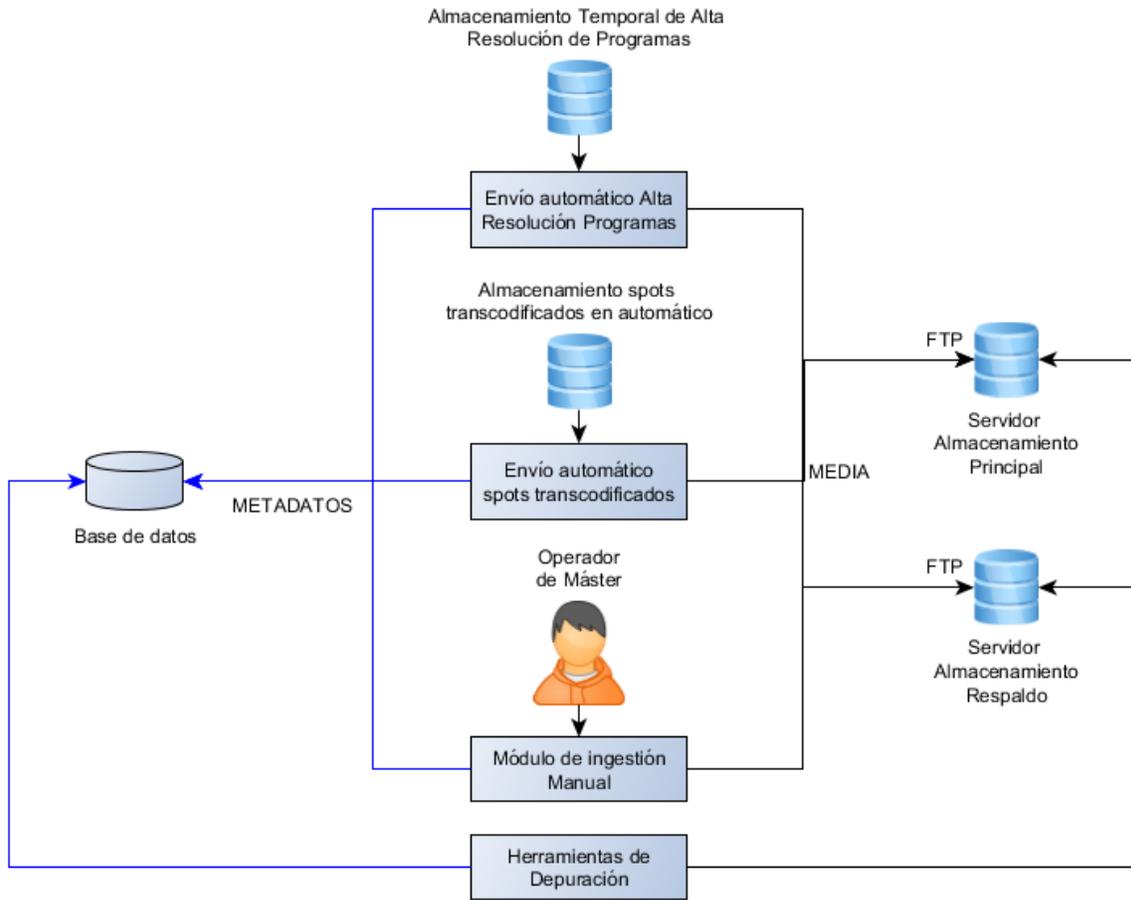


Figura 7 Módulos del subsistema de almacenamiento

Para desarrollar del subsistema de almacenamiento se identificaron los siguientes elementos:

- Diseño de la base de datos del subsistema de almacenamiento: Creación de los objetos en la base de datos necesarios para almacenar los datos que permitieran la gestión de los contenidos dentro del sistema.
- Módulo envío automático de alta resolución de programas: Componente encargado de enviar los archivos resultantes del proceso de transcodificación de programas en forma automática a los servidores de almacenamiento.
- Módulo envío automático de spots transcodificados: Realiza el envío de los archivos resultantes del proceso automático de recodificación de spots a los servidores de almacenamiento.
- Módulo de ingestión manual: Se ha expuesto ya que mediante este módulo se procesan los contenidos de tipo súper, bloques individuales de programa y spots, sin embargo este módulo también cuenta con interacción con el subsistema de almacenamiento porque incluye la funcionalidad para enviar estos materiales a los servidores de almacenamiento.
- Instalación, configuración y pruebas de los dos servidores de almacenamiento.
- Herramientas de depuración: Se plantearon además tres herramientas de depuración, la primera encargada de borrar los contenidos de las producciones televisivas que han sido enviados mediante los módulos de ingestión WEB. La segunda herramienta de depuración se encarga de borrar los materiales que corresponden a los programas que no son transcodificados en automático. La tercera herramienta sincroniza el espacio real disponible en los servidores con los registros correspondientes en la base de datos.

3.5.1 Base de datos del subsistema de almacenamiento.

La mayor parte de la base de datos del sistema de automatización se encuentra contenida en la tabla TVP_MAT_TRANS, en esta tabla como ya se ha mencionado se almacenó la información de cada uno de los materiales de video que se utilizarían en la transmisión. De esta manera cada uno de los registros que están en la tabla representa un archivo de media que en algún momento será enviado al subsistema de almacenamiento. Además la tabla cuenta con los campos suficientes para determinar si el registro ya cuenta con un archivo de video asignado o bien si aún no tiene media en el subsistema de almacenamiento.

La estructura de la tabla es la siguiente:

Tabla 10. Estructura tabla base de datos TVP_MAT_TRANS

Campo	Tipo	Propósito
CODIGO_MAT	Char(11)	Identificador del material utilizado solamente dentro del sistema de automatización, por default es (0) si el registro es nuevo y no tiene material físico asignado aún
FECHA	DATE	Fecha de alta en el subsistema de almacenamiento
ID_VID	Number(4)	Número consecutivo que lleva la cuenta de videos recibidos en un día
CTE_ID	Varchar2(40)	Cliente al que pertenece el material
VER_DESCRIP	Varchar2(50)	Descripción del material
VER_DUR	Number(10)	Duración planeada del material en segundos
VERSION	Varchar(25)	Nombre de la versión del material
TAM_ARCH	Number(15)	Tamaño del archivo en bytes
DURACION	Number(8)	Duración real del material en milisegundos
VER_NO_INV	Varchar2(16)	Código de inventario de 16 dígitos para bloques de programa y 8 dígitos para spots.
ID_ESP_SERV_PRIN C	Number(2)	Id del espacio de almacenamiento donde se aloja el material en el servidor de almacenamiento principal.
ID_ESP_SERV_RESP	Number(2)	Id del espacio de almacenamiento donde se aloja el material en el servidor de almacenamiento de respaldo.

La tabla guarda el registro de cada uno de los archivos existentes en el subsistema de almacenamiento así como los que aún están a la espera de material audiovisual, como

podemos observar también tiene referencia a los espacios de almacenamiento dentro de los servidores en el cual está alojado el material.

El concepto de espacio de almacenamiento se definió como el disco duro dentro del servidor de almacenamiento en el cual se encuentra el archivo de media asignado al registro. En servidor de almacenamiento se instalaron 12 discos duros dedicados al almacenamiento de los videos, cada disco tiene una capacidad de 4TB, teniendo así disponible en total una capacidad de almacenamiento por servidor de 48 TB. Esta capacidad de almacenamiento fue diseñada para almacenar alrededor de 5 años de producciones televisivas y spots que pueden ser descargados en automático por los reproductores y ocuparlos en la transmisión.

El mecanismo mediante el cual los reproductores se conectan a los servidores y la estructura de archivos del sistema de almacenamiento serán descritos más adelante.

Para almacenar la información relativa a los espacios de almacenamiento se creó una tabla de en la base de datos con el nombre TVP_ESPACIOS_ALMACENAMIENTO, en esta tabla es donde se crean los identificadores que se manejan en la tabla TVP_MAT_TRANS.

La estructura de la tabla TVP_ESPACIOS_ALMACENAMIENTO es la siguiente:

Tabla 11. Estructura tabla base de datos TVP_ESPACIOS_ALMACENAMIENTO

Campo	Tipo	Propósito
ID	Number(3)	Identificador único del espacio de almacenamiento
ID_Servidor	Number(2)	Identificador del servidor al que pertenece el espacio de almacenamiento
Ruta_servidor	Varchar2(256)	Ruta local dentro del servidor
Ruta_compartido	Varchar2(256)	Ruta de la ubicación en las unidades compartidas del servidor
Espacio_total	Integer	Espacio total del espacio de almacenamiento
Espacio_ocupado	Integer	Espacio ocupado del espacio de almacenamiento
Ruta_FTP	Varchar2(256)	Alias de la ubicación en el servidor FTP del servidor de almacenamiento
Orden	Number(3)	Índice del espacio de almacenamiento, indica el orden en el que se deben ir ocupando

Las dos tablas descritas contienen toda la información necesaria para administrar el almacenamiento de los archivos de video en los servidores de almacenamiento.

El sistema guarda una copia de cada archivo en cada uno de los servidores en un esquema de un servidor principal y uno de respaldo, de esta forma el subsistema de almacenamiento tiene redundancia en caso de que el servidor principal falle. En este tipo de sistemas de automatización, los cuales prestan servicio ininterrumpido, es muy importante contar con redundancia en caso de algún fallo.

La estructura de la tabla TVP_MAT_TRANS fue heredada de otras interfaces entre el sistema de tráfico y los sistemas de automatización. Se decidió dejar la tabla sin cambios para agilizar el desarrollo de los componentes y no tener que hacer cambios significativos en los módulos involucrados. También se tomó en cuenta que el sistema de tráfico tiene acceso a esta tabla para realizar acciones de alta de registros y si se introducían cambios también en ese sistema tendrían que ser soportados para no tener ningún problema.

3.5.2 Módulo envío automático de alta resolución de programas

El módulo de transcodificación del subsistema de ingestión entrega como salida los archivos en alta resolución correspondientes a cada bloque de programa del capítulo ingestado, esto se expuso en la sección dedicada a ese subsistema. Los archivos de alta resolución como se explicó quedan almacenados temporalmente dentro del servidor de ingestión, desde esa ubicación el módulo de envío automático de alta resolución de programas detecta la presencia de nuevos archivos y se encarga de enviarlos utilizando el protocolo FTP a los servidores de almacenamiento.

Para realizar el envío el módulo realiza esta serie de pasos con cada bloque de programa encontrado en la carpeta donde los deposita el módulo de transcodificación:

1. Búsqueda del registro del capítulo al que pertenece el bloque de programa en la tabla TVP_CAPITULOS para obtener los datos requeridos en el proceso de envío a servidores de almacenamiento. Esto es posible realizarlo debido a que cada archivo contiene en el nombre el identificador del registro del capítulo además del bloque de programa al que pertenece el material.
2. Cálculo de un nuevo código de material para este archivo. El código de material es un identificador único para cada archivo de video en el sistema. Es un número de 11 dígitos de longitud. Los dos primeros son un identificador del sistema de automatización al que pertenece, 6 más para la fecha en el que se envía el archivo al subsistema de almacenamiento y 3 que son un número consecutivo. El consecutivo se reinicia cada día en 1 y cada archivo cuenta con un número consecutivo diferente que va indexando los archivos de manera secuencial enviados al subsistema de almacenamiento en un mismo día.
3. El archivo a enviar es renombrado con el código de material obtenido
4. En el paso 1 se obtiene el sub código de inventario, a este identificador se le agregan los 2 dígitos que indican el número de bloque para conformar un código de inventario completo. Con este código de inventario se revisa si ya existe registro en la tabla TVP_MAT_TRANS. En el caso que ya exista el código de

inventario en la tabla se actualiza el código de material del registro así como los campos de la fecha, el consecutivo, la duración real y el tamaño del video. Si el registro aún no existe se crea con todos los campos que debe tener un registro.

5. Búsqueda del espacio de almacenamiento con suficiente capacidad para almacenar el archivo. Esto depende del servidor al cual se va enviar el material, primero se manda al servidor principal y una vez terminado el proceso se regresa a este paso para enviarlo al servidor de respaldo. La búsqueda del espacio se apoya en la tabla ESPACIOS_ALMACENAMIENTO que guarda la información del espacio total y el ocupado en cada disco.
6. El archivo se envía mediante FTP al espacio de almacenamiento encontrado en el paso anterior.
7. Si el envío es exitoso se actualiza el campo del servidor correspondiente en el registro de la tabla TVP_MAT_TRANS y además suma el espacio recién ocupado al registro del espacio de almacenamiento correspondiente en la base de datos en la tabla ESPACIOS_ALMACENAMIENTO.
8. Si está pendiente el envío del archivo al servidor de respaldo entonces va al paso 5 buscando un espacio de almacenamiento adecuado en el servidor de respaldo. En caso contrario no hay otro servidor al cual enviar y el proceso termina.

Una vez que el archivo de video se encuentra depositado en los servidores ya está disponible para ser descargado y transmitido si es necesario. El módulo de envío de alta resolución se encuentra activo de forma ininterrumpida revisando si hay nuevos materiales resultantes del proceso de transcodificación.

Desarrollé este módulo utilizando lenguaje C++ con la integración de clases para el manejo del protocolo FTP desarrolladas en el framework del área. El tiempo de desarrollo y pruebas fue de 1 semana. La aplicación se instaló en el servidor de ingestión ocupando una interfaz de red diferente a la dedicada para recibir el material, esto se logró mediante la edición de las tablas de ruteo en el sistema operativo.

En cada uno de los servidores de almacenamiento se instaló un servidor FTP FileZilla en el que está configurado un usuario de FTP con acceso a todos los discos de almacenamiento mediante un alias para cada uno de ellos como se encuentra registrado en la tabla ESPACIOS_ALMACENAMIENTO, de esta manera el módulo utiliza los datos adecuados del alias registrado en la tabla para establecer la conexión FTP y en consecuencia depositar el archivo en la ruta correcta.

3.5.3 Módulo envío automático de spots transcodificados

De forma similar al proceso anterior, el resultado de la transcodificación automática de spots son los archivos en alta resolución de los spots, los cuales se depositan en una ubicación dentro del equipo donde se ejecuta el proceso, dicho equipo es el servidor al cual se envían los spots mediante el módulo de ingestión WEB de spots. El módulo de envío automático de spots transcodificados se instaló para ser ejecutado simultáneamente

en este equipo, el módulo fue diseñado para detectar si existen archivos nuevos como resultado de la transcodificación y se encarga de enviarlos a los servidores de almacenamiento.

El proceso de envío sigue una serie de pasos similares al módulo de envío de bloques de programa, los cuales son los siguientes para cada spot:

1. El nombre del archivo en alta resolución contiene el código de inventario del spot, el cual se busca en la tabla TVP_MAT_TRANS. Si existe un registro con el código de inventario correspondiente y no tiene material asignado el proceso puede continuar, en caso contrario termina rechazando el material ya que el registro debe estar libre de material para efectuar el proceso, de otra forma sería ambiguo saber si el usuario que envió el material se equivocó de código de inventario o si desea cambiar la versión del material almacenada en el sistema.
2. Cálculo de un nuevo código de material para este archivo.
3. El archivo a enviar es renombrado con el código material determinado en el paso anterior.
4. En la tabla TVP_MAT_TRANS se actualiza el código de material del registro con el código de inventario correspondiente así como los campos de la fecha, el índice consecutivo, la duración real y el tamaño del video.
5. Búsqueda del espacio de almacenamiento con suficiente espacio para almacenar el archivo. Al igual que en módulo anterior, primero se manda al servidor principal y una vez terminado el proceso se regresa a este paso para enviarlo al servidor de respaldo. La búsqueda del espacio se apoya en los datos almacenados en la tabla ESPACIOS_ALMACENAMIENTO para determinar cuál tiene el espacio suficiente.
6. El archivo se envía mediante FTP al espacio de almacenamiento encontrado.
7. Si el envío es exitoso se actualiza el campo del espacio de almacenamiento correspondiente en el registro de la tabla TVP_MAT_TRANS y suma el espacio recién ocupado al registro del espacio de almacenamiento en la base de datos.
8. Si está pendiente enviar el archivo al servidor de respaldo entonces va al paso 5 buscando espacio en el servidor de respaldo. En caso contrario no hay otro servidor al cual enviar y el proceso termina.

El proceso realizado por este módulo es similar al módulo dedicado al envío de programas, sin embargo la diferencia principal es que está dedicado a archivos de material de spot y además los registros de los códigos de inventario en la tabla TVP_MAT_TRANS deben existir por anticipado, estos debieron darse de alta en el sistema de tráfico con anterioridad y éste a su vez ingresa el registro en la tabla.

El módulo para envío de spots también lo desarrollé utilizando lenguaje C++ integrando las librerías de FTP desarrolladas por el equipo de trabajo. La duración del desarrollo incluyendo las pruebas fue de alrededor de 3 días. Participé realizando el diseño, implementación y pruebas de este módulo.

3.5.4 Módulo de ingestión manual

El módulo de ingestión manual se creó para ingresar el material de tipo súper así como bloques individuales de programa y spots, lo cual se expuso en la sección anterior. Este módulo también cuenta con la funcionalidad para enviar mediante FTP los materiales a los servidores de almacenamiento. De hecho este módulo se desarrolló en un proyecto anterior al sistema de automatización de máster en el cual también participé. Los módulos de envío automático son la una versión derivada de este módulo de operación manual.

Al utilizar el módulo de ingestión manual todos los materiales deben contar con un registro previo en la tabla TVP_MAT_TRANS. El módulo inmediatamente después de importar un archivo o crear un súper determina un código de material nuevo y en la tabla TVP_MAT_TRANS el registro correspondiente es actualizado con ese código de material así como los campos de duración, tamaño del archivo y fecha de envío. Entonces el archivo de media es renombrado con el código de material creado y en este punto se encuentra solamente en el equipo donde se ejecuta el módulo de ingestión manual en una ubicación configurable destinada para un almacenamiento temporal.

Los materiales tienen que ser enviados al subsistema de almacenamiento, se definió por regla general que todo material debería estar guardado en los servidores de almacenamiento para que pueda ser descargado por los reproductores que lo requieran. El módulo de ingestión manual cuenta con la funcionalidad necesaria para enviar los materiales mediante FTP a los servidores de almacenamiento con un proceso similar al realizado por el módulo de envío automático de spots, pero solamente incluye los pasos a partir de la búsqueda del espacio de almacenamiento adecuado, enviando primero el material al servidor principal y luego al servidor de respaldo.

Realicé el diseño, desarrollo e implementación de este módulo, la adecuación de características para el envío automático en este módulo requirió un tiempo de 3 días incluyendo las pruebas. El módulo fue desarrollado en lenguaje C++ e incluye la clase para el manejo de FTP implementadas en el framework del equipo de desarrollo.

3.5.5 Herramientas de depuración del subsistema de almacenamiento

El espacio de almacenamiento es un recurso finito que al paso del tiempo se agotaría porque siempre se estarían ingesting nuevos materiales en el sistema de automatización, por lo anterior se definieron procesos de depuración diseñados para liberar el espacio ocupado por las producciones televisivas cuya transmisión ha concluido o bien materiales que ya no se van a ocupar. Para realizar la depuración del subsistema de almacenamiento se desarrollaron tres herramientas que se deben ejecutan de manera periódica para garantizar la existencia de espacio libre.

La primera herramienta es la aplicación depuración de capítulos completos ingestados. En la tabla TVP_PROGRAMAS dentro de la base de datos, se tiene un campo llamado depuración, este campo es una bandera. Si el campo contiene el valor de 0 significa que

la depuración para esa producción no está activa, por el contrario si el valor está en 1 la depuración por medio de esta aplicación está activa para esa producción. La herramienta debe ser ejecutada manualmente en periódicamente en intervalos de tiempo entre 15 días y máximo una vez al mes para realizar la depuración y tomar lectura del nivel de espacio libre en los servidores de almacenamiento. La ejecución de la herramienta sigue los siguientes pasos para efectuar la depuración:

1. Búsqueda en la tabla de programas de las producciones televisivas donde la bandera de depuración está activada.
2. Recorre uno a uno los programas encontrados buscando en la tabla de capítulos todos los registros aún disponibles para esa producción eligiendo los registros que tienen una fecha de transmisión que sea menor a la fecha actual menos 7 días , es decir, fechas anteriores a una semana a partir de la fecha actual.
3. Con la lista de capítulos que cumplen el criterio y son candidatos para la depuración se consulta la tabla de los bloques de programa tomando como base el identificador de cada capítulo, en esta tabla hay un campo que contiene el código de material de cada bloque, de esta forma se obtiene la lista de códigos de material pertenecientes al capítulo que se está procesando.
4. Recorre la lista de bloques por cada capítulo obtenida en el paso anterior.
5. Con el código de material de cada bloque obtenido en el paso anterior se busca su registro en la tabla TVP_MAT_TRANS obteniendo el identificador de los espacios de almacenamiento tanto en el servidor principal como en el servidor de respaldo en los cuales está guardado el archivo de media.
6. La aplicación accede a la ubicación del espacio de almacenamiento correspondiente en el servidor principal y borra el material, además se anota en la bitácora la operación y se lleva el conteo total de espacio desocupado.
7. La aplicación accede a la ubicación del espacio de almacenamiento correspondiente en el servidor de respaldo y borra el material, además se anota en la bitácora la operación y se lleva el conteo total de espacio desocupado real en el servidor de respaldo.
8. En este punto ya no existen los materiales físicamente en los servidores, entonces se pueden ir borrando las referencias de la base de datos. La primera tabla es TVP_MAT_TRANS.
9. Se borra el registro del bloque de programa en la tabla de bloques.
10. Si hay más bloques en la lista regresa al paso 4, de lo contrario continúa.
11. Al terminar de procesar todos los bloques es posible borrar el capítulo de la tabla de capítulos.
12. Si hay más capítulos para procesar regresa al paso 3, de lo contrario continúa.
13. Si ya no se tienen más capítulos por procesar para esa producción pasa a la siguiente producción para buscar sus capítulos regresando al paso 2. Si no hay más producciones candidatas el proceso se da por terminado presentando un resumen con el conteo de los bloques candidatos para borrado, el conteo de los materiales borrados físicamente y la cantidad de espacio liberado en el proceso.

Para definir las producciones que son candidatas para el proceso de depuración se consulta a los programadores de las guías y en la base de datos dentro de la tabla TVP_PROGRAMAS se actualiza el campo de las producciones aprobadas para ser depuradas. La información acerca del espacio ocupado en cada espacio de almacenamiento se almacena en una tabla en la base de datos, el sincronizar el espacio real ocupado con el registro de la tabla en base de datos es una tarea reservada para la tercera herramienta de depuración que será expuesta más adelante en esta sección.

Realicé en el diseño, desarrollo e implementación de esta herramienta, el código está escrito en lenguaje C++ y el tiempo de desarrollo fue de 1 semana. Sin embargo el desarrollo de la herramienta de depuración se llevó a cabo después de la puesta en marcha del sistema en una segunda fase de desarrollo.

La segunda herramienta es la herramienta de depuración de materiales individuales, esta herramienta fue diseñada para borrar todos aquellos materiales que no ingresen mediante la Interfaz Web de Ingestión de programas. La ejecución esta herramienta también es manual y se ejecuta regularmente cada mes para eliminar los materiales que han llegado por la herramienta de ingestión manual, la ingestión de spots o la ingestión de bloques de programa individuales.

Existen producciones televisivas cuya emisión es única, por ejemplo los noticieros, los eventos deportivos, programas de noticias del espectáculo o especiales de fin de año. Este tipo de contenidos generalmente no se vuelven a transmitir, se transmiten solo una vez en cierta fecha o bien tienen una única retransmisión. La herramienta da acceso a los registros de los materiales que no han sido ingestados mediante la Interfaz de Ingestión de Programas, para ello se apoya en la tabla de los bloques de programas y la tabla TVP_MAT_TRANS, todos aquellos registros que no se encuentran en la tabla de bloques de programa pero si en la tabla TVP_MAT_TRANS son los candidatos para la depuración de esta herramienta, este cruce se realiza con base en el campo de código de material.

La herramienta permite seleccionar los registros que se desean depurar, busca los espacios de almacenamiento correspondientes tanto en el servidor principal como en el servidor de respaldo, borra los archivos físicamente de los servidores y elimina las referencias de la tabla TVP_MAT_TRANS.

De esta forma estas dos herramientas fueron diseñadas para ser complementarias y así contar con la depuración de todo tipo de materiales independientemente de la vía mediante la cual fueron ingestados. Se definió que para conocer la lista de los materiales que pueden ser eliminados del sistema mediante la herramienta de depuración de materiales individuales se debe consultar a los usuarios programadores de guías. Esta herramienta la desarrollé en lenguaje C++ un tiempo de 3 días incluyendo pruebas. La herramienta de depuración de materiales individuales también fue parte de la segunda fase de desarrollo una vez que se tenía el sistema en producción.

La tercera herramienta que fue desarrollada para llevar a cabo la depuración del subsistema de almacenamiento se llama sincronización de espacios de almacenamiento,

se diseñó con la función de actualizar los registros de la tabla de espacios de almacenamiento con la información del espacio libre en los discos de los servidores.

Esta herramienta también se ejecuta de forma manual preferentemente al terminar de utilizar alguna de las dos herramientas anteriores y se ejecuta en cada servidor de almacenamiento. El funcionamiento interno incluye cargar la lista de registros de espacios de almacenamiento de la tabla en la base de datos correspondiente al servidor donde se está ejecutando la aplicación, después con la información obtenida busca en el sistema de archivos los discos indicados en la lista de registros para determinar el espacio libre en cada uno de ellos, finalmente actualiza los registros de la tabla de espacios de almacenamiento guardando exactamente el espacio ocupado en cada disco.

La actualización de los registros de espacios de almacenamiento es una tarea que podría haberse implementado dentro de las otras dos herramientas de depuración, sin embargo se decidió tener un diseño modular en el sistema, de esta forma se pueden crear en el futuro diferentes herramientas que permitan realizar la depuración de los materiales o bien crear procesos que puedan ser operados por los usuarios y la herramienta de sincronización sería igualmente útil e incluso sería posible automatizar su ejecución.

La herramienta de sincronización de espacios de almacenamiento la desarrollé en lenguaje de programación C++. Como las otras dos herramientas expuestas también fue incluida en una segunda fase de desarrollo una vez que el sistema ya se encontraba transmitiendo la señal de los canales para los cuales fue creado. El tiempo de desarrollo fue de 2 días incluyendo las pruebas necesarias para validar su funcionamiento.

3.6 Módulo de grabación

Una de las necesidades solicitadas para sistema de automatización fue la capacidad de grabar contenidos que son transmitidos en vivo en otros canales o bien señales provenientes de algún estudio en vivo para su posterior retransmisión. La operación de guardar una señal, generalmente de una fuente en vivo, y reproducirla tiempo después para su transmisión es llamada diferir el contenido.

El Módulo de grabación de contenido se planteó para cubrir parte de esta necesidad, como su nombre lo indica fue desarrollado para realizar la captura de señal de televisión en banda base, esto es señal sin compresión que llega de los estudios u otras señales de la televisora donde se despliegue el sistema. La señal es capturada mediante hardware especializado para grabación y reproducción de señal de televisión en banda base.

El hardware disponible para este proyecto fueron tarjetas marca BlackMagic Design modelo Decklink Quad. BlackMagic Design es una empresa internacional que fabrica una gran variedad de productos para el procesamiento de audio y video tanto para empresas televisivas como para largometrajes y postproducción. En el 2002 lanzaron al mercado el primer modelo de la tarjeta Decklink, la cual posteriormente dio lugar a una familia de productos diseñados para la grabación y reproducción de señales de televisión. Los

productos de este fabricante generalmente se caracterizan por ser de buena calidad a precios muy accesibles dentro de la industria de la televisión.

La tarjeta Decklink Quad pertenece a una familia de tarjetas PCI Express dedicadas a la captura y reproducción de señales sin compresión, permiten manejar señales en formatos SD y HD, incluso es capaz de manejar formatos en 3D y UltraHD. El modelo Decklink Quad se compone de 4 tarjetas SDI para captura o reproducción, y cada una se puede utilizar de manera independiente.

SDI son las siglas de Serial Digital Interface, es una familia de interfaces de video digital utilizada principalmente en la transmisión de señal de video profesional sin compresión y sin encriptación. La interfaz está diseñada para su operación en distancias cortas hasta de 30 metros. Las conexiones físicas se realizan mediante cables coaxiales y conectores BNC que pueden transportar señales tanto en HD como en SD. Este tipo de interfaz es capaz de utilizar palabras de datos de 8 o 10 bits y es capaz de integrar la transmisión de audio conjuntamente con el video con hasta 16 canales de audio

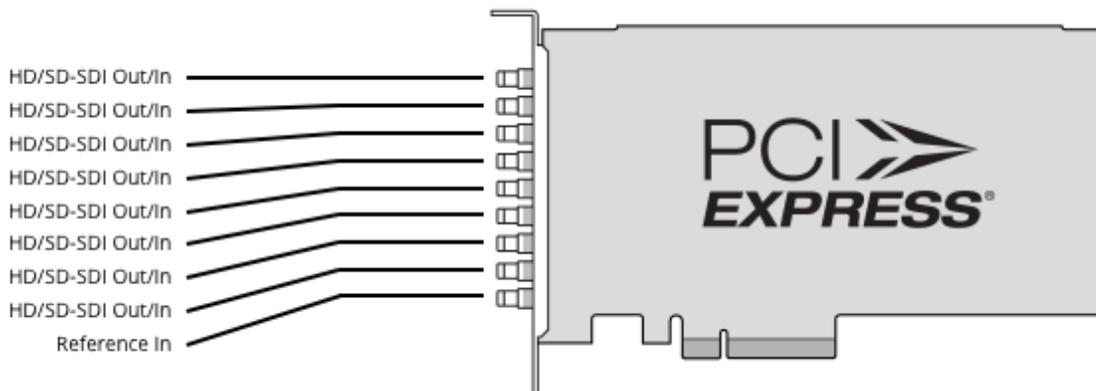


Figura 8 Diagrama de conexiones de la tarjeta BlackMagic Decklink Quad

Como se observa en el diagrama de conexiones de la figura 8, tomado de Blackmagic Design (2017), la tarjeta cuenta con 8 conectores disponibles para ser configurados ya sea como entradas o salidas y un conector más para la entrada de una señal de referencia. La señal de referencia es una señal de reloj que permite sincronizar varios equipos de transmisión de tal manera que las transiciones entre ellos al aire se realicen de manera exacta. En el caso de este modelo las señales de referencia aceptadas son Try-Sync y BlackBurst. Este modelo tiene la capacidad para manejar formato de video en SD bajo los estándares NTSC y PAL o bien en formato HD a diferentes resoluciones y framerate. El muestreo del audio soportado es el comúnmente utilizado de manera profesional en la industria de la televisión, esto es 48 kHz y una resolución de 16 bits por muestra. También cuenta con la capacidad de realizar mediante hardware conversiones entre formatos ya sea de formato HD a SD o de formato SD a HD tanto en sus señales de entrada como en las señales de salida.

La tarjeta además cuenta con un SDK disponible para lenguaje de programación C++, este SDK se puede usar bajo la tecnología de Microsoft DirectShow o bien a bajo nivel integrando las bibliotecas con aplicaciones escritas en C++ permitiendo comunicarse directamente con el driver del dispositivo. Para la implementación de los diferentes componentes del sistema de automatización que requieren utilizar esta tarjeta desarrollé una clase en lenguaje C++ que permite la comunicación con el driver pero a un nivel más intuitivo. Algunas de las funciones que permite realizar esta clase son las siguientes:

- Elegir un dispositivo Decklink del equipo para comunicarse con su driver
- Configurar el dispositivo en modo de captura o reproducción
- Configurar el formato de captura o reproducción del dispositivo con el que se está trabajando.
- Arrancar el dispositivo para el modo elegido
- Si el dispositivo se arranca en modo captura permite al driver comunicarse con la aplicación para que ésta última pueda recibir los datos resultantes de dicha captura.
- Si el dispositivo arranca en modo reproducción la clase permite enviar a la tarjeta los datos de audio y video para la transmisión a la salida de los mismos.
- Detener el dispositivo ya sea en modo captura o reproducción
- Monitorear el estado de la tarjeta.
- Liberar el driver dispositivo para que otra aplicación pueda comunicarse con ese dispositivo.

Utilizando el paradigma de programación orientada a objetos cada instancia de la clase entonces representa un dispositivo con el cual podemos trabajar. De hecho es posible utilizar los cuatro dispositivos de la tarjeta Decklink Quad simultáneamente o bien si se tienen dos tarjetas Decklink Quad en el equipo incluso manejar los 8 simultáneamente creando en la aplicación un objeto para cada tarjeta.

El trabajo con este tipo de hardware se basa en el intercambio de buffers o matrices de bytes que contienen la información de audio o video que se desea enviar a la tarjeta para su reproducción o bien en el caso de realizar una captura recibir datos de la tarjeta.

Cada buffer de video representa un cuadro o frame de la señal de video, y se maneja en un espacio de color denominado UYVY o 4:2:2.

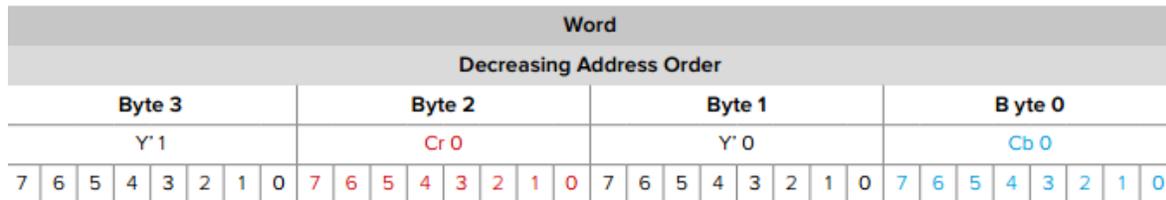


Figura 9 Direccionamiento de espacio de color UYVY en memoria

En la figura 9, tomada de Blackmagic Design (2017), observamos que el espacio de color UYVY utiliza 4 componentes cada una de 1 byte y palabras de 32 bits o 4 bytes, las componentes se acomodan dentro del buffer de la siguiente manera:

En este espacio de color dos píxeles se almacenan en 32 bits o 4 bytes, la información del color o crominancia correspondiente a los dos píxeles se almacena en 2 bytes y la luminancia de cada uno en los 2 bytes restantes de la palabra. La cantidad de bytes necesaria para almacenar cada renglón de la imagen o línea horizontal se obtiene multiplicando el ancho de la imagen por el número de bytes necesario para cada pixel, en este caso por 2. La cantidad de bytes necesarios para almacenar un cuadro completo se determina multiplicando los bytes por renglón por el número de renglones de la imagen o la resolución a lo alto de la imagen. En el caso de imágenes HD en resolución 1920 x 1080 entonces tenemos 3840 bytes en cada renglón y cada cuadro de imagen se almacena en 4147200 bytes, alrededor de 4MB. Para imágenes SD con resolución 720 x 480 se manejan 1440 bytes por cada renglón y cada cuadro de imagen requiere 691200 bytes, lo cual es alrededor de 675 KB de espacio.

El espacio de color recibe la denominación UYVY debido a las componentes que maneja. En la componente U (o Cb0 en la figura) se almacena parte de la crominancia o información relativa al color correspondiente a ambos píxeles de la palabra, en la componente V (Cr0 en la figura) está almacenada la segunda componente de crominancia correspondiente a ambos píxeles de la palabra. Las componentes Y (Y'0 y Y'1 en la figura) son la información de la luminancia o dicho de otra forma qué tan brillante es ese pixel, a diferencia de la crominancia cada una de estas componentes corresponden respectivamente al primer pixel y segundo pixel almacenados en los 4 bytes de la palabra.

Este formato fue creado en las tecnologías análogas que permitieron el desarrollo de la televisión a color, está basado en un modelo de percepción de información en el ojo humano. Según las investigaciones el ojo humano cuenta con más receptores sensibles a la luminosidad de los objetos que observamos, es decir la información con la que se puede construir una imagen “en blanco y negro”, y cuenta con número menor de receptores que son sensibles a los colores de los objetos, es decir la información de color o crominancia. De ahí que en este espacio de color exista más información relativa a la luminancia que a la crominancia.

El espacio de color UYVY no cuenta con compresión, estamos hablando de un formato en banda base, sin embargo fue ideado para ser más eficiente que un formato por ejemplo RGB en donde cada pixel ocupa 3 bytes y por lo tanto una pareja de pixeles se almacenaría en 6 bytes. Puede parecer poca la diferencia pero la cantidad de pixeles que se manejan tanto en formato HD como en SD hace que se tenga una gran ventaja almacenarlos en UYVY en vez de otro espacio de color. El hardware compatible con este formato es capaz de interpretar la información contenida en estas componentes y generar toda la gama de colores de una imagen para que el ser humano pueda observarla adecuadamente y con claridad.

Según expuesto en el buffer de bytes para el intercambio de información con la tarjeta, el primer byte contiene la información de la primera componente de crominancia de la primer pareja de pixeles, el segundo byte la luminancia del primer píxel de esta pareja, en el tercer byte la segunda componente de crominancia de la pareja de pixeles y el cuarto byte la luminancia del segundo pixel de la pareja. A partir del quinto byte pasamos a la segunda pareja de pixeles de la imagen con el mismo acomodo de componentes y así contiguamente en el buffer de almacenamiento hasta llegar al final de los bytes destinados para un renglón de pixeles de la imagen. Al terminar los bytes del renglón inmediatamente se almacenan en los siguientes bytes la información de los pixeles del siguiente renglón de la imagen siguiendo el acomodo descrito. El buffer debe ser de elementos de tipo BYTE o unsigned char cuando se desarrolla en lenguaje C++ para almacenar valores que van del 0 a 255 en cada componente, al modificar los valores de las componentes almacenados en el buffer es posible modificar la imagen que se va a capturar o reproducir.

En este hardware el audio de la señal también se maneja a través del intercambio de información a través de buffers. La tarjeta BlackMagic Decklink Quad permite capturar o reproducir audio en la señal ya sea con 2, 8 o 16 canales independientes. Este hardware no permite trabajar con otra combinación de número de canales como 5, 6 ó 10 sino solamente las que se mencionaron.

La resolución del audio que maneja la tarjeta es de 16 bits, es decir una muestra de un canal se almacena en 2 bytes. Cada muestra debe ser almacenada entonces para cada canal que vamos a reproducir o capturar, de manera que para almacenar una muestra se necesitan 2 bytes multiplicados por el número de canales que contiene la señal. La forma de almacenar dentro del buffer la información es tal que toda la información de la primera muestra con todos sus canales se almacena en los primeros bytes, al terminar ese segmento se almacena la información de la segunda muestra con todos sus canales y así sucesivamente hasta contener todo un paquete de muestras.

Por ejemplo si estamos manejando audio a dos canales el buffer internamente contiene la información de la siguiente forma:

Tabla 12. Almacenamiento de muestras de audio en memoria

Byte	Muestra	Almacena	Canal
0	1	8 bits menos significativos	1
1	1	8 bits más significativos	1
2	1	8 bits menos significativos	2
3	1	8 bits más significativos	2
4	2	8 bits menos significativos	1
5	2	8 bits más significativos	1
6	2	8 bits menos significativos	2
7	2	8 bits más significativos	2

Como podemos observar en la tabla se almacenan contiguamente en la memoria las muestras de audio, cada muestra ocupa 2 bytes ya que su resolución es de 16 bits, por lo tanto cada muestra tiene un rango de valores con signo que van de -32767 a 32768 lo cual es ideal debido a la naturaleza senoidal de una señal de audio. Si tenemos 2 canales necesitamos 4 bytes para almacenar una muestra contemplando ambos canales, si tenemos 8 canales entonces son 16 bytes para almacenar 1 muestra y para 16 canales necesitaríamos 32 bytes. El audio que maneja este hardware es sin compresión y se puede calcular la cantidad de datos necesaria para almacenar la información en cierta cantidad de tiempo, el cálculo se puede hacer multiplicando el número de canales por 2 bytes por muestra por el número de muestras por segundo que son 48000 en cada segundo. Por ejemplo para 2 canales serían 192000 bytes por segundo y para 8 canales 768000 bytes por segundo. Los cálculos anteriores también nos dan la idea de cuánta información debe procesar la aplicación por segundo y debe ser lo más rápido posible para que no llegue a una condición de sobreflujo en la operación.

Los buffers de datos que he mencionado hasta el momento no tienen ningún tipo de encabezado o información de algún protocolo, los datos almacenados en los bytes representan directamente pixeles de video o muestras de audio que provienen de entradas de señal en banda base o bien se van a transmitir a la salida.

En el sistema de automatización el módulo de grabación se desarrolló para capturar los contenidos provenientes de los estudios y otras fuentes que pueden contener hasta 6 canales de audio por lo que la tarjeta se configura para realizar la captura de la señal con 8 canales de audio, internamente la aplicación toma las muestras correspondientes a los canales del 1 a 6 y la información correspondiente a los canales 7 y 8 es descartada para no almacenar información que no se va a ocupar posteriormente. En cuanto al video el cliente solicitó que las señales a capturar fueran solamente en formato HD aunque el

módulo también se diseñó con la capacidad de grabar en SD si se requiere, lo cual se puede definir a través de los archivos de configuración.

El SDK de la tarjeta requiere una serie de pasos para inicializar el hardware a utilizar lo cual incluye la elección del dispositivo, el formato de la señal y elección del modo de captura o reproducción. La comunicación con el dispositivo para el intercambio de información de audio y video se lleva a cabo mediante “callbacks”, esto es una llamada a una función definida por el programador en una aplicación, la cual se realiza cuando otra función es ejecutada. El mecanismo de callbacks permite manejar diferentes capas de abstracción y además permite la ejecución de acciones asíncronas. Gran cantidad de hardware que he integrado para el procesamiento de señales tiene un funcionamiento basado en este modelo de comunicación. Funciona de una manera sencilla, el programador define una función que será invocada cuando el driver ejecute la función que tiene definida para atender cierto evento, por ejemplo al capturar un cuadro de video, o cuando se termina de reproducir un cuadro de video o si detecta un cambio de formato en la señal de entrada. Generalmente en este mecanismo el programador debe registrar de alguna manera la función de “callback”, en el momento de inicializar el hardware, la cual será llamada desde la función ejecutada por el driver al ocurrir el evento para el cual fue definida, además se deben respetar los argumentos definidos en la documentación del SDK así como el tipo de retorno, es decir la definición de su prototipo.

Una vez que está registrada la función de callback se puede arrancar el dispositivo en el modo de captura y a partir de este punto el driver entregará a la aplicación mediante las llamadas de callback la información correspondiente a la señal capturada, para esta tarjeta en particular en la misma llamada se entregan dos buffers separados representados como dos argumentos de la función de callback, el primero contiene la información del cuadro de video capturado, es decir la imagen, y el segundo el paquete de muestras del audio correspondiente.

El diagrama de bloques del módulo de grabación es el siguiente:

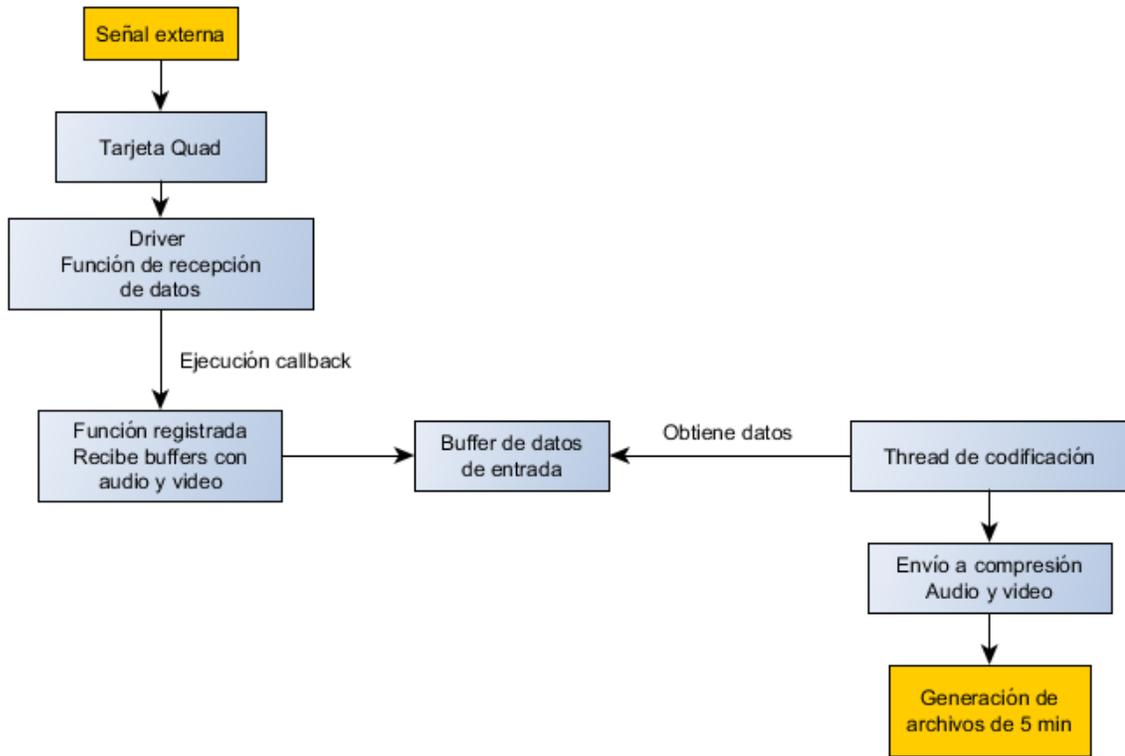


Figura 10 Diagrama de bloques de módulo de grabación de señales diferidas

El módulo fue diseñado de forma que a través de la función registrada como callback recibe los datos de audio y video mediante buffers de bytes éstos son enviados a su vez a un buffer de datos de entrada. El buffer de datos de entrada es un arreglo de estructuras de datos que permiten almacenar una copia del buffer de audio y video recibidos, el buffer es en sí mismo una estructura de datos tipo cola o FIFO (primero que entra primero que sale). La función de este buffer de datos es evitar la pérdida de datos ante posibles interrupciones en el proceso de codificación o bien de otras tareas en el sistema operativo que puedan afectar el desempeño de la aplicación. Otra manera de describirlo es como el mecanismo de “antishock” disponible en los reproductores de discos compactos, principalmente portátiles, que surgieron en la década de los 90’s, estos mecanismos guardaban en un buffer una cantidad de datos de audio decodificados para que ante posibles interrupciones en el proceso de lectura, el proceso de reproducción no viera comprometido a diferencia de las primeras versiones de esos dispositivos.

En el caso del módulo de grabación este buffer permite que el driver siga entregando datos a la aplicación no importando que el proceso de codificación pueda ser interrumpido por otros procesos o el sistema operativo, el tamaño del buffer se determinó de forma experimental para que no se desborde en el caso de esas interrupciones y puede

almacenar hasta 2 segundos de contenido capturado. La velocidad de ejecución de los procesos en los equipos en los cuales fueron desplegados es suficiente para procesar la información que captura la tarjeta en tiempo real, de otra forma el buffer siempre se desbordaría y la grabación no sería de alta calidad porque perderíamos información que sería descartada, en otras palabras tendríamos una condición de sobreflujo. Los buffers de entrada y de salida, como veremos en el módulo reproductor, son parte fundamental en este tipo de sistemas para garantizar tanto la grabación como reproducción fluida y de alta calidad.

En el módulo se incluyó un thread dedicado a la lectura de los datos del buffer de entrada y la compresión del audio y el video. Es un proceso separado del hilo principal de ejecución de la aplicación para que las operaciones de la interfaz gráfica no interfieran con su operación. El thread tiene integrada las librerías de libav, en específico la sección que contiene las funciones para realizar la compresión de los datos de audio y video, así como para depositarlos en archivos digitales. Dentro de la configuración del módulo se define en qué ruta se depositan los archivos resultantes de la compresión. El módulo fue diseñado para operar las 24 horas del día de manera que los archivos se generan en segmentos exactos de 5 minutos cada uno. La salida del módulo de grabación en este punto se compone de archivo de audio y un archivo de video por separado. El video es codificado con el códec H264 al igual que los demás videos almacenados en el sistema de automatización, sin embargo en este caso se trata de una grabación de alta calidad cuyo bitrate es de 30 Mbps y solamente incluye cuadros Intra.

En los estándares de codificación MPEG de manera interna se manejan varios tipos de imagen, entre los cuales tenemos 3 principales:

- Imágenes I o Intra: Imágenes cuya codificación depende de sí mismas. No tienen referencias a otras imágenes. Se utiliza la mayor cantidad de bits para codificarlas y se puede empezar a reproducir un video desde este punto sin problema ya que contiene toda la información necesaria para recuperar la imagen.
- Imágenes P o Predicted: Son predichas por extrapolación, es decir se generan a partir de una referencia a una imagen I o P anterior en el tiempo. A través de estimaciones de movimiento se extrapolan a partir de las otras imágenes de referencia.
- Imágenes B o Bi-predicted: Son imágenes que se generan a partir de la interpolación de otras dos de referencia. Las imágenes de referencia pueden ser de tipo I o P y deben ser una anterior en el tiempo y una posterior. Generalmente este tipo de imágenes son las que requieren menor cantidad de información para ser almacenadas ya que son interpoladas y por lo tanto requieren el mínimo de información.

La siguiente figura expone las relaciones entre estos tipos de imágenes:

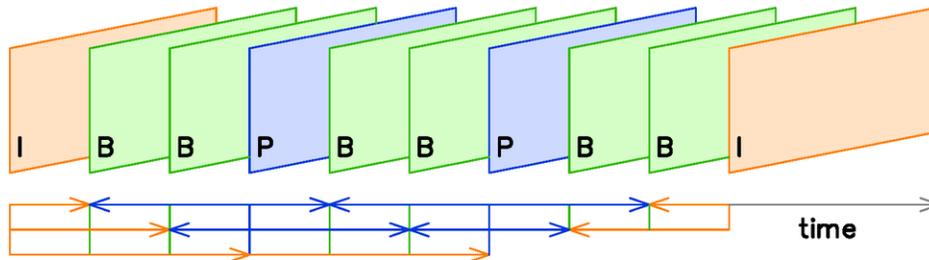


Figura 11 Relación de imágenes I,B y P en el estándar MPEG

En la familia de estándares de compresión MPEG se cuenta con un concepto llamado GOP o Group of pictures (Grupo de imágenes) que define la combinación de este tipo de imágenes o cuadros dentro de la secuencia del video. El GOP cuenta con una propiedad llamada longitud que indica la distancia en imágenes entre cuadros I, en el caso de la figura sería un GOP de longitud 9. Además se indican cuántos cuadros P existen en el GOP y cuántos cuadros B se encuentran entre las imágenes I o P. En el caso del ejemplo serían 2 imágenes P y 2 cuadros tipo B. Es importante conocer el concepto del GOP para manipular correctamente los parámetros de compresión, entre más largo sea el GOP los puntos de referencia para iniciar la reproducción estarán más espaciados en tiempo, estos cuadros pueden utilizarse como puntos de resincronización y corresponden a los cuadros I. En términos prácticos podemos observar el efecto de lo anterior en los receptores de televisión, cuando hay una falla y aparecen pixelizaciones en la imagen o “digitalizaciones” pasa un tiempo para que el receptor pueda recuperar la calidad en la imagen lo cual sucede cuando encuentra un cuadro I y es capaz de reconstruir de nuevo la imagen completa, los otros tipos de imágenes son muy susceptibles a las pérdidas de información. Sin embargo entre más cuadros B o P tengamos entonces tendremos una imagen de mayor calidad en menos espacio necesario para su almacenamiento, es decir estará más comprimida. Debido a los puntos mencionados entonces el módulo de grabación se diseñó para codificar el video solo en cuadros I y así poder realizar cortes precisos posteriormente sin tener la necesidad de recodificar el video para regenerar los cuadros de otro tipo, ya que si intentamos cortar un video en un cuadro B o P tendríamos que decodificar los necesarios para reconstruirlos y posteriormente recodificar la señal, esto también explica por qué el bitrate es superior para estos archivos ya que al ser solamente imágenes I necesitamos mayor espacio de almacenamiento al tener una menor tasa compresión de la señal y por lo tanto conservar una buena calidad de imagen.

El audio se captura sin compresión y es guardado un archivo wav, en el cual almacena las muestras de audio en PCM. Los archivos de audio tipo wav incluyen un encabezado en que se indica el número de canales que contiene el archivo, la resolución de cada muestra y la duración del audio, entre otras propiedades de la señal de audio.

Los archivos generados de video y audio quedan en una ubicación temporal en el sistema, posteriormente un thread de multiplexado incluido en este módulo toma la pareja de archivos correspondientes y crea a partir de estos archivos un único archivo en formato mp4 que contiene audio y video sincronizados. El archivo resultante es depositado en una ubicación final desde donde se puede acceder a este material a través del módulo de extracción de material grabado si se requiere.

El módulo de grabación de señales fue programado en lenguaje C++ con la integración de la tarjeta Decklink Quad mediante la clase que desarrollé para la comunicación con su driver, por lo tanto se utilizó el SDK de la tarjeta BlackMagic Decklink. Para las tareas de codificación y multiplexado se integró parte de la biblioteca libav. Participé directamente en el desarrollo del módulo tanto en la programación como en pruebas e instalación. El tiempo de desarrollo fue de dos semanas. El módulo se diseñó para funcionar de manera ininterrumpida las 24 hrs capturando las señales externas que le sean suministradas. El equipo en el que se instaló tiene la capacidad para procesar y almacenar grabaciones de 3 señales externas de manera simultánea en formato HD. Las grabaciones se guardan hasta por 3 días, después de los cuales los archivos de video son depurados por la siguiente herramienta.

3.6.1 Herramienta de depuración automática del módulo de grabación

La capacidad de almacenamiento de los equipos siempre es finita, en el caso del módulo de grabación está diseñado para almacenar los videos capturados de hasta tres días, tiempo suficiente para extraer el material y almacenarlo en los servidores del subsistema de almacenamiento para su posterior uso en la señal al aire.

Desarrollé una herramienta de depuración programada en lenguaje C++ que se encarga de monitorear los archivos generados por el módulo de grabación, si esta aplicación detecta archivos que fueron capturados hace más de 3 días los elimina, de esta forma la operación del módulo de grabación no se interrumpe en ningún momento por falta de espacio de almacenamiento. Esta es una herramienta muy sencilla que se instaló en el mismo equipo de grabación y fue desarrollada en 1 día de trabajo incluyendo las respectivas pruebas. A pesar de su sencillez, la tarea que ejecuta es de vital importancia para que el módulo de grabación cumpla con su labor. Esta herramienta funciona de manera autónoma las 24 horas del día y cuenta con un tiempo entre revisiones de archivos configurable mediante un archivo de parámetros de configuración, además como todas las demás aplicaciones es capaz de guardar una bitácora con las acciones más importantes que ejecuta.

3.7 Módulo de extracción de grabaciones

El módulo de grabación descrito en el apartado anterior se encarga de generar segmentos de videos 5 minutos ininterrumpidamente. La siguiente etapa en el proceso exigía la capacidad suficiente para que el usuario seleccionara fragmentos de la señal capturada con la finalidad de extraer los programas de las grabaciones, los cuales serían ocupados para su transmisión en la señal de salida del sistema de automatización.

El módulo de extracción de grabaciones se desarrolló para realizar esta tarea, fue diseñado con una interfaz gráfica a manera de línea de tiempo con intervalos de 5 minutos, la cual refleja el resultado de la captura de la señal por parte del módulo de grabación de señales indicando si los videos ya se encuentran listos. El usuario debe posicionarse entonces en el bloque de 5 minutos que contiene el inicio del fragmento que necesita extraer, una vez que lo ha ubicado puede seleccionar mediante arrastre todos los bloques de 5 minutos que contienen el programa que el usuario necesita.

En el módulo se integró la tecnología de Apple QuickTime, la cual permite la reproducción de archivos de media con audio y video, además QuickTime cuenta con un SDK en lenguaje de programación C con cual es posible realizar tareas en la aplicación como selección de fragmentos de media, exportar el material seleccionado a un archivo independiente en alguna unidad de almacenamiento con la que cuente el equipo, acceso a los datos de audio y video, creación de archivos, edición básica, entre otras funcionalidades incluidas en el SDK.

Una vez que el usuario ha seleccionado todos los segmentos que contienen el programa a extraer del módulo de grabación se muestra una vista previa del video en un reproductor donde el usuario puede elegir los puntos de entrada y salida de su selección con precisión de cuadros. La integración de QuickTime hace posible realizar el marcado de la selección así como la exportación de la misma a un archivo en la ubicación que desee el usuario. Una de las características de diseño en el módulo de grabación cobra entonces sentido en la exposición de este módulo, ya que grabar el audio sin compresión y el video en cuadros I el corte puede ser preciso sin la necesidad de recodificar, acelerando el proceso. En otras aplicaciones es posible cortar en medio de la estructura del GOP sin embargo ese proceso requiere de la recodificación del material al realizar el corte porque de otra manera se vería comprometida la reproducción al momento de decodificar la secuencia si se ha cortado en un cuadro que depende de la interpolación de algún otro y esos cuadros de referencia son simplemente eliminados de la selección. El corte de videos que contienen cuadros B y P requiere por lo tanto más tiempo y capacidad de cómputo, lo cual se contrapone con una de las características que fueron solicitadas para el módulo: realizar el proceso en la menor cantidad de tiempo posible.

Una vez que ha sido extraído el programa o bloque de programa necesario a un archivo, éste último se ingesta al sistema de automatización a través del módulo de ingestión manual donde el usuario cuenta con la lista de los códigos de inventario y puede encontrar el registro que pertenece al material extraído. Por lo anterior es que en el momento de cerrar la guía de programación se insertan los registros de los materiales aún

no entregados en la tabla TVP_MAT_TRANS, de otra forma el usuario no contaría con los registros necesarios para ingestar estos materiales al sistema.

El módulo de ingestión manual ya ha sido detallado, en dicho módulo se genera el código de material al importar el archivo de media y es enviado a los servidores de almacenamiento finales quedando disponible para los reproductores.

El módulo de extracción de grabaciones se configuró montando como unidades compartidas en red las ubicaciones donde el módulo de grabación deposita los archivos capturados en el equipo donde se instaló, de esta manera el módulo de extracción tiene acceso a las grabaciones. El módulo de extracción de materiales fue diseñado para tener varias instancias funcionando simultáneamente e incluso en equipos independientes, por ello se comparten las unidades del equipo encargado de la grabación.

Los módulos de grabación cuentan con redundancia para minimizar las fallas ocasionadas tanto por el hardware como en cualquier otra eventualidad. Se instaló un equipo principal con capacidad para grabar 3 señales como se mencionó en la sección anterior y un equipo de respaldo con la misma capacidad. Cada módulo de extracción de materiales es capaz de acceder a las grabaciones del equipo principal o de respaldo en automático.

Participé realizando el diseño y programación del módulo de extracción de grabaciones en lenguaje C y C++. En este módulo se integró el SDK de QuickTime para realizar las operaciones de exportación, selección y la reproducción de la vista previa del material elegido. El tiempo de desarrollo fue de 2 semanas incluyendo las pruebas necesarias para asegurar la calidad del software.

3.8 Módulo de reproducción

El último módulo desarrollado para el sistema de automatización de máster es el módulo de reproducción. Este módulo recibe los resultados de todas las etapas anteriores del sistema y los utiliza para generar una señal de televisión en banda base ininterrumpida, es por lo tanto el módulo más complejo del proyecto.

El módulo de reproducción debido a su complejidad y las tareas que desempeña cuenta con dos componentes:

- Servicio de interfaz con el hardware
- Reproductor de máster

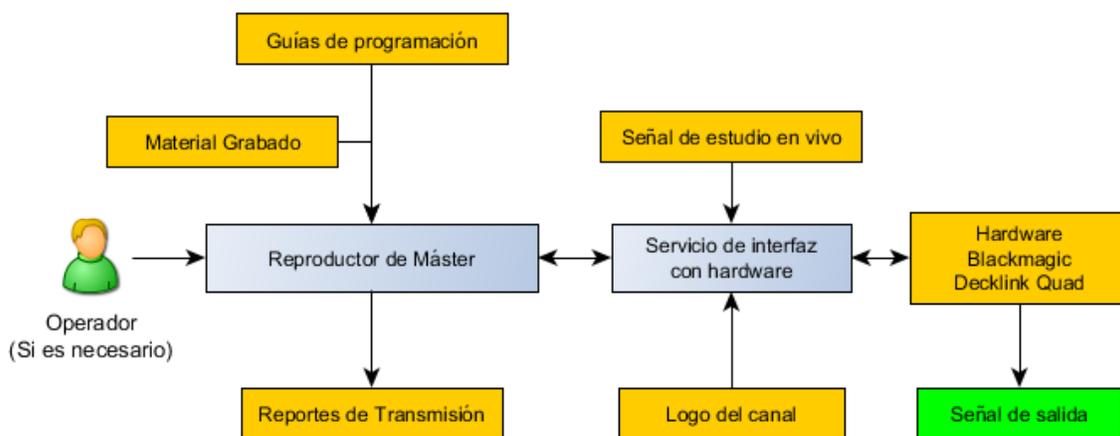


Figura 12 Diagrama de funcionamiento del módulo de reproducción

3.8.1 Servicio de interfaz con el hardware

El servicio de interfaz con el hardware es el componente del software encargado del control del hardware destinado para generar la señal de televisión y el envío de la información tanto de audio como de video al hardware para su transmisión en la salida del sistema. Fue diseñado para ejecutarse bajo una plataforma Windows en la forma de un servicio del sistema operativo, se decidió así porque el sistema operativo es capaz de asignar una alta prioridad a este tipo de procesos, lo cual es ideal en este caso porque este componente se encarga de la comunicación directa con el driver del hardware. Por otro lado este tipo de servicios se pueden configurar para arrancar junto con el sistema operativo y finalmente el desarrollar una capa de software destinada a la comunicación con el hardware nos permite utilizar este componente en otras aplicaciones que necesitan realizar reproducción de audio y video a través del hardware utilizado.

Las tareas del servicio son las siguientes:

- Control y manejo del hardware
- Recepción de información proveniente del reproductor de máster
- Envío de audio y video al hardware
- Recepción de la señal de estudio y envío de la misma al hardware de salida si se requiere
- Inserción de cuetones
- Inserción de logo en la señal de salida
- Inserción de sùpers en la señal de salida
- Manejo de salida en formato HD y SD simultáneo, si es necesario
- Normalización automática del nivel de audio
- Control del mecanismo de protección bypass

Para el desarrollo del servicio se tomaron en cuenta una serie de consideraciones en el diseño entre las cuales tenemos las siguientes:

- Aplicación multithread. Debido a la cantidad de operaciones a realizar se necesitaron varios hilos de ejecución que nos permitieran ejecutar tareas de forma simultánea.
- Comunicación interproceso. Al ser una aplicación con varios threads, éstos se deberían comunicar en algún momento con un proceso principal ya sea para sincronizar operaciones o bien notificar las actividades que están realizando. En este servicio la comunicación interproceso se implementó a través mensajes de Windows.
- Buffers de datos. Como se ha expuesto anteriormente el manejo de audio y video requirió almacenar esta información en conjuntos de bytes o buffers, la aplicación debe ser capaz de guardar varios paquetes de audio y cuadros video para evitar pérdida de información debida a posibles variaciones en los tiempos de ejecución o bien sincronización con los elementos del hardware.
- Manejo de comunicación entre aplicaciones. El módulo de reproducción se implementó dividido en dos aplicaciones, por lo que es necesario tener algún mecanismo de comunicación entre ambas, el cual se detalla más adelante.
- Manejo de Callbacks. La interacción con el hardware así como la comunicación interproceso se diseñaron con un esquema de callbacks, similar a lo descrito en la sección del módulo de grabación de señales.

El diseño y arquitectura del servicio se muestra en la siguiente figura:

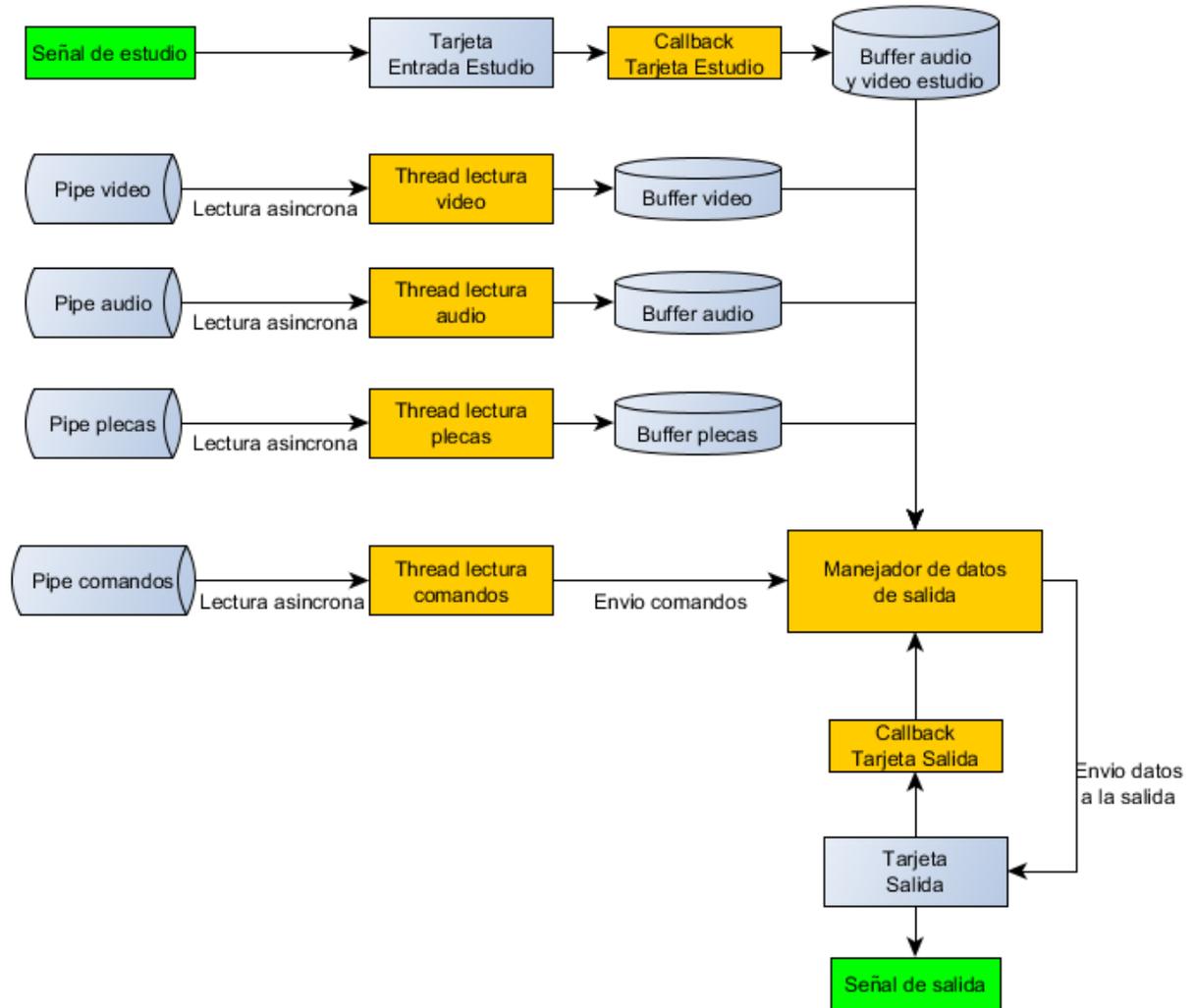


Figura 13 Arquitectura del servicio de interfaz con el hardware

Control y manejo del hardware

El hardware destinado para este proyecto fueron tarjetas BlackMagic Decklink Quad, el mismo modelo utilizado en el módulo de grabación, las principales características ya fueron detalladas. Para el desarrollo del servicio se utilizó la misma clase de interacción con el hardware escrita en C++ que se expuso en la sección del módulo de grabación. Dicha clase incluye las funciones necesarias tanto para captura como para reproducción.

La clase desarrollada en el caso del módulo de reproducción nos permite la comunicación con el hardware en un modelo orientado a objetos para realizar las siguientes funciones:

- Elección de los dispositivos a utilizar para la captura de señales de estudio
- Elección de los dispositivos para la reproducción
- Configuración del formato y características en el dispositivo de captura o reproducción
- Inicialización del dispositivo para captura o reproducción
- Comunicación con el hardware a través de una función de callback para recibir los datos de la captura de señal del estudio.
- Envío de información de audio y video al hardware para reproducción
- Comunicación con el dispositivo mediante una función de callback para conocer el estado de la reproducción y envío de más datos a la salida.

El servicio utiliza la clase para controlar los dispositivos necesarios de la tarjeta Decklink Quad, para lo cual se crea un objeto de esta clase por cada dispositivo con el que necesita interactuar. El intercambio de datos, como ya se expuso en la sección del módulo de grabación, se diseñó tanto para captura como para reproducción a través de buffers de bytes donde la información corresponde a cuadros de vídeo en espacio de color UYVY y el audio se maneja en formato PCM, es decir sin compresión, con una frecuencia de muestreo de 48kHz y una resolución de 16 bits.

El servicio como las demás aplicaciones del sistema cuenta con un archivo de configuración que nos permite definir su comportamiento y los dispositivos que va a utilizar. La tarjeta BlackMagic Decklink Quad incluye cuatro dispositivos independientes que se pueden configurar para reproducción o captura, el servicio cuenta con soporte para generar la señal de salida por uno o dos dispositivos simultáneamente, además la captura del estudio se puede realizar por otro dispositivo, estos dispositivos se pueden definir en el archivo de configuración.

Al iniciar la ejecución del servicio se inicializan los dispositivos elegidos para captura o reproducción, los pasos son descritos en la documentación del SDK del fabricante.

Para iniciar la reproducción en un dispositivo BlackMagic Decklink Quad se debían seguir los siguientes pasos:

1. Seleccionar el dispositivo
2. Configurar el formato para la reproducción de audio y video
3. Activar la salida de video

4. Activar la salida de audio
5. Configurar la función de callback para notificaciones de cuadros reproducidos en la salida de la tarjeta
6. Activar el mecanismo de pre-roll para el audio, un pre-roll significa preparar datos para su reproducción.
7. Enviar audio y video para el pre-roll, esta información se almacena en un buffer interno del hardware que es una cola de reproducción. La cantidad de cuadros debe ser suficiente para garantizar la reproducción continua y la menor posible para no elevar la latencia del sistema. Actualmente el nivel de pre-roll es de 15 cuadros o aproximadamente medio segundo tanto en audio como video.
8. Iniciar la reproducción de audio y video
9. Mientras la reproducción se encuentra activa el driver reporta a través de la llamada a la función de callback el término de la reproducción de cada cuadro, en ese instante el servicio debe enviar un cuadro más de video con su audio correspondiente a la tarjeta para ser formada en la cola de reproducción del hardware. De esta manera el nivel de la cola se mantiene y el hardware cuenta con la información suficiente para continuar la reproducción.

En el caso de la captura para recibir la señal de un contenido en vivo el hardware exigía los siguientes pasos:

1. Seleccionar el dispositivo para captura
2. Configurar el formato de captura para audio y video
3. Activar la entrada de video
4. Activar la entrada de audio
5. Configurar la función de callback mediante la cual el hardware reportará que ha capturado un nuevo cuadro.
6. Iniciar la captura.
7. Mientras la captura no se detenga cada que el hardware haya capturado un nuevo cuadro y lo tenga listo para entregarlo al servicio lo reportará mediante la llamada de la función registrada para el callback.

Tanto la captura como la reproducción se diseñaron para ser configuradas en formato de video HD con una resolución de 1920x1080 pixeles y una tasa de 29.97 cuadros por segundo o en formato de video SD con una resolución de 720x480 pixeles a una tasa de 29.97 cuadros por segundo.

En cuanto al audio, en la captura y en la reproducción se manejaron siempre 8 canales con un muestreo de 48kHz y una resolución de 16 bits. El audio se manejó siempre a 8 canales porque la señal de los estudios en vivo podía traer 6 canales de audio y por otro lado en los canales 7 y 8 se solicitó como parte de los requerimientos que fueran insertados los cuetones. La explicación de qué es un cuetone se detalla más adelante.

La configuración habitual para el sistema de automatización de máster que fue solicitada incluía una salida de video en formato HD y otra salida simultánea de video en SD con el mismo contenido debido a que los diferentes formatos de salida tenían como destino

diferentes cadenas de distribución de televisión tanto en México como otros mercados internacionales. La entrada de estudio se configuró únicamente a través de una sola tarjeta de entrada y con un equipo externo llamado *botonera* se podía conmutar entre 16 señales de estudio diferentes a seleccionar. El servicio de comunicación con el hardware no tenía la limitante de manejar solo una entrada de video, se diseñó para operar con incluso con 2 tarjeta Decklink Quad, tomando en cuenta que se requieren dos señales de salida podríamos tener hasta 6 señales de entrada, sin embargo por costos y debido a que ya se contaba con la *botonera* se eligió tener una sola señal de entrada.

Recepción de información proveniente del reproductor de máster

El sistema de automatización fue desarrollado para utilizar dos fuentes de información para la reproducción, la primera son los archivos de media almacenados en el disco duro de los servidores de almacenamiento, en otras palabras es material grabado. La segunda fuente de información es la señal en vivo proveniente de los estudios que se captura como ya se describió a través del hardware.

Más adelante se expone a detalle el componente del sistema encargado de decodificar el audio y video del material grabado, sin embargo cabe mencionar en este punto que la aplicación dedicada a esa tarea es el Reproductor de Máster.

En ejecución, el Reproductor de Máster y el servicio de interfaz con el hardware son dos procesos separados ejecutados dentro del mismo equipo de cómputo, dado que los videos se decodifican en el reproductor de máster extrayendo el audio y video de los mismos y el servicio es el componente encargado de enviar esta información al hardware ambos procesos se deben comunicar de alguna manera para enviar los datos de un proceso a otro. Existen diferentes mecanismos de comunicación interprocesos que ofrecen los sistemas operativos, para este proyecto que ejecuta bajo el sistema operativo Windows se eligió el mecanismo de pipes de datos.

Un pipe de datos en el sistema operativo Windows, como se describe en Microsoft (2018), es una sección de memoria compartida que usan los procesos para comunicarse. El proceso que crea el pipe es llamado servidor del pipe. El proceso que se conecta a un pipe es un cliente del pipe. En este mecanismo de comunicación interproceso un proceso escribe información en el pipe y posteriormente el otro proceso lee la información desde el pipe. La lectura y escritura pueden ser en un sentido, es decir solo un extremo del pipe escribe y el otro solamente lee, o bien ser bidireccional en cuyo caso ambos extremos pueden leer y escribir. Además en Windows se tienen los pipes anónimos o sin nombre y los pipes con nombre.

Los pipes anónimos siempre tienen un solo sentido y se utilizan típicamente para comunicar procesos padres con procesos hijos, otra de sus características es que siempre son locales, es decir los procesos ejecutan en el mismo equipo. Los pipes con nombre pueden tener un solo sentido de comunicación o ser duplex y permiten la comunicación entre el servidor del pipe y uno o varios clientes, en este último caso todas las instancias deben manejar el mismo nombre y cada proceso tiene sus propios buffers e

identificadores dentro del sistema operativo. Los pipes con nombre pueden usarse en aplicaciones cliente-servidor en red. Cualquier proceso puede conectarse a un pipe con nombre, atendiendo a las reglas de seguridad en Windows, lo que hace de este mecanismo una forma sencilla de comunicación entre dos procesos independientes.

Para comunicar el reproductor con este servicio se eligieron entonces los pipes con nombre que permiten la comunicación entre dos procesos independientes. Los pipes son relativamente fáciles de manejar debido a que las funciones para lectura y escritura son las mismas que ofrece el SDK de Windows para llevar a cabo la lectura y escritura de archivos. Otra de las características atractivas de los pipes es que las operaciones de lectura y escritura pueden ser síncronas o asíncronas, lo cual nos permite ya sea detener la ejecución de un proceso hasta que la operación termina o bien continuar haciendo otras operaciones hasta que el sistema operativo nos avise que se ha terminado la operación a través de un mecanismo de callback.

Como lo muestra la arquitectura del servicio, el diseño incluyó 4 pipes:

- Pipe de audio: Canal de comunicación para recibir el audio de los videos decodificados por el reproductor.
- Pipe de video: Pipe dedicado al envío de video decodificado por el reproductor.
- Pipe plecas: Las plecas o supers también son decodificadas en el reproductor y cada cuadro es enviado al servicio para superponerlo en la señal de salida.
- Pipe de comandos: El reproductor fue diseñado para llevar el control de la guía de programación, transiciones de eventos y otras operaciones. De tal forma que el módulo de reproducción también actúa como un módulo de control para el servicio del hardware que le indica las operaciones que debe realizar y en qué momento.

Para atender estos 4 canales de comunicación se crearon 4 procesos hijos del servicio en los cuales se inicializan los pipes, es decir son los extremos servidores de los pipes, y realizan operaciones de lectura para recibir la información depositada en los pipes.

Cada uno de estos hilos de ejecución tiene un ciclo de actividades para realizar la lectura de información de los pipes como el siguiente:

- Revisa que haya un cliente conectado en el otro extremo del pipe, de lo contrario el ciclo de lectura no continúa y espera otra revisión.
- Si hay un cliente conectado revisa que la cantidad de datos sea la esperada dependiendo de la información que se recibe por el pipe, si la cantidad en bytes es la adecuada continua, de lo contrario debe esperar otra iteración del ciclo.
- Realiza la operación de lectura en forma asíncrona registrando una función de callback al término de la misma.
- Recibe la notificación del término de la operación de lectura mediante la función de callback además de una bandera que indica el éxito de la operación, si la operación fue exitosa procesa la información dependiendo de la naturaleza de la misma que puede ser audio, video, cuadro de un súper o un comando dependiendo del pipe y el thread.

Thread de video

El thread de video se diseñó para recibir cada cuadro de video proveniente de la decodificación realizada en el reproductor de Máster. Los cuadros están codificados en espacio de color UYVY. En el buffer de datos enviado además de la información del cuadro de video lleva un encabezado con información relevante del cuadro entre la que se encuentra la siguiente:

- Ancho del cuadro en pixeles
- Alto del cuadro en pixeles
- Número de cuadro dentro del video decodificado
- Total de cuadros del archivo de video que está siendo decodificado desde el reproductor de máster.
- Formato de la imagen o espacio de color
- Un identificador único de control de decodificación que representa el archivo de video de donde proviene el cuadro

Con la información anterior el thread es capaz de realizar la lectura de cada cuadro de video a través del ciclo antes descrito y validar que la información del encabezado es adecuada. Si la información es correcta entonces el cuadro se guarda en un buffer de video. El buffer de video es una cola circular que nos permite almacenar hasta 2 segundos de video para prevenir cualquier pérdida de información que pudiera presentarse debido variaciones en los tiempos de ejecución ocasionadas por ejemplo por interrupciones del sistema u otras razones. Más adelante expondré el uso de este buffer.

Thread de audio

El thread de audio se diseñó de forma similar al thread de video, es decir se encarga de recibir paquetes de audio provenientes del proceso de decodificación del reproductor de máster. Cada paquete de audio contiene las muestras correspondientes al tiempo de despliegue de cada cuadro de video. Recordemos que el framerate de video en la televisión es 29.97 cuadros, para alcanzar esa cantidad de paquetes y lograr reproducir sincronizado el video con el audio el cual tiene un muestreo de 48Khz, los paquetes contienen un número variable de muestras, que pueden ser 1600, 1601 o 1602. El número de muestras por paquete de audio se controla en el proceso de decodificación del reproductor de máster. Esta secuencia de número de muestras es conocida en el ámbito del procesamiento digital de audio para la televisión y existen varios modelos de secuencias para alcanzar la tasa de muestreo utilizada profesionalmente.

De manera similar a los paquetes de video, cada buffer en el cual se envía un paquete de audio además de los bytes que contienen las muestras de audio tiene un encabezado con la siguiente información:

- Identificador único de control de decodificación que representa el archivo del cual proviene el paquete de audio
- Formato del audio, planeado para soporte de otros formatos no solo PCM
- Número de canales del paquete
- Resolución del audio en bytes
- Número de muestras del paquete
- Frecuencia de muestreo
- Número de paquete dentro del total que tiene el archivo origen
- Número total de paquetes de audio en el archivo origen

El thread revisa que cada paquete de audio recibido contenga las características adecuadas en el encabezado para ser enviado a la tarjeta de salida, sin embargo no es enviado directamente a la tarjeta de salida sino a un buffer temporal que puede almacenar también hasta 2 segundos de audio en paquetes, esto nos previene contra pérdidas de información.

Otra tarea del thread de audio consiste en obtener el promedio de los valores RMS de cada muestra en el paquete recibido con la finalidad de conocer el nivel de audio, posteriormente esta información se utiliza para realizar el proceso de normalización del nivel de audio. Cada estructura que contiene un paquete de audio guardada en el buffer temporal contiene este valor para ser aprovechado en ese proceso posterior.

Thread de plecas

Una característica solicitada para el sistema de automatización de máster es la posibilidad de reproducir plecas o súpers, los cuales son secuencias de imágenes que se superponen en la señal de salida y cubren una porción de la pantalla. Los cuadros de video de estos contenidos pueden tener transparencia y animación. Los archivos almacenados en el sistema provienen de secuencias de imágenes en formato TGA con 4 componentes de color en formato RGB más un canal de transparencia llamada Alpha Channel, por lo tanto cada pixel se almacena en 32 bits. Estas secuencias de imágenes recordemos que se procesan con el módulo de ingestión manual.

La aplicación de Reproducción de Máster se diseñó para decodificar los archivos almacenados en formato MOV y extraer cada cuadro de la animación en su espacio de color original, es decir RGBA. Cada cuadro decodificado es enviado al servicio de interfaz con el hardware donde se recibe en el thread de plecas a través de un pipe de datos. De manera similar al thread de video cada buffer utilizado para enviar un cuadro de súper contiene un encabezado con la siguiente información:

- Ancho del cuadro en pixeles
- Alto del cuadro en pixeles
- Número de cuadro dentro del súper decodificado
- Total de cuadros del súper que está siendo decodificado desde el reproductor
- Formato de la imagen o espacio de color, para futuro soporte de otros formatos actualmente solo está soportado el RGBA

- Un identificador único de control de decodificación que representa al súper de donde proviene el cuadro

Al igual que en los otros procesos, el thread verifica si la información recibida en el encabezado es válida. Si la información es correcta entonces el cuadro es transformado de espacio de color RGBA a espacio de color UYVY para facilitar la operación de combinación con la señal de fondo que ya está en formato UYVY, además se almacenan en una matriz los coeficientes del canal de transparencia para cada pixel. Se desarrolló una optimización en el proceso de combinación con base en un arreglo que contiene el mismo número de banderas que el alto en pixeles de la imagen, la bandera por renglón se enciende si alguno de los píxeles contenidos en el renglón tiene el alpha channel activo, de esta forma se puede descartar el procesamiento de cierto renglón si no es necesario combinarlo. Este formato lo diseñé para realizar la conversión de cada cuadro del súper en un proceso independiente al de salida, de tal forma que estos cálculos se realizan dentro del thread de recepción y las operaciones propias de la combinación con la imagen de fondo se hacen en el subproceso que envía el video al dispositivo de salida. El resultado de la conversión de cada cuadro por lo tanto tiene un arreglo en espacio de color UYVY, un arreglo del alpha channel a cuadro completo y además un arreglo de banderas contenido una por cada renglón.

El formato no existe en la industria de la televisión, fue diseñado para el sistema de automatización, al cual que he llamado UYVY con canal alpha. La información se envía a un buffer temporal donde se van almacenando los cuadros de los súper en estructuras de datos para su posterior combinación con la señal de salida. El buffer de las plecas es una cola circular que nos permite almacenar hasta 2 segundos de cuadros de plecas.

Thread de comandos

El servicio de interfaz con el hardware es el componente del sistema diseñado para enviar información de audio y video a los dispositivos de salida de la señal, capturar señal de estudio y enviarla a la salida, entre otras tareas. Sin embargo el orden de las instrucciones y el momento preciso en el cual deben ser ejecutadas se controla en la aplicación Reproductor de Máster. Por lo tanto el Reproductor de Máster rige el comportamiento del servicio de interfaz con el hardware indicando las operaciones que debe realizar y el momento en el cual debe hacerlas.

Las instrucciones que el Reproductor de Máster puede enviar al servicio de interfaz con el software se llaman comandos. Los comandos viajan también a través de un pipe de datos, el servicio de interfaz con el hardware cuenta con un thread dedicado a las operaciones de lectura del pipe de comandos. Los paquetes que viajan por este pipe tienen una longitud de 32 bytes, contienen las instrucciones y los datos necesarios para ejecutarlas. El thread de comandos se encarga de leer los paquetes con los comandos del pipe, interpreta su contenido y ejecuta la instrucción modificando la señal de salida o realizando las acciones necesarias dependiendo del comando recibido.

Los comandos soportados son:

- Iniciar de reproducción
- Detener reproducción
- Iniciar el envío de señal estudio a la salida
- Finalizar el envío de señal estudio a la salida
- Iniciar el pre-roll de señal de estudio
- Activar logo
- Desactivar logo
- Insertar cue tone de entrada
- Insertar cue tone de salida

Los threads expuestos anteriormente son subprocesos diseñados para estar en operación durante todo el tiempo de ejecución del servicio, cuentan con un ciclo que revisa si existen datos en los pipes y realiza su lectura, este ciclo se mantiene activo todo el tiempo porque los datos enviados mediante el mecanismo de pipes pueden llegar en cualquier momento y siempre deben ser extraídos de los pipes. Sin embargo aunque la ejecución del ciclo no termina se hace una pausa en cada iteración del ciclo para darle oportunidad a los demás procesos del sistema de ocupar tiempo de procesador.

El manejo de todos los pipes en los threads incluye la realización de operaciones de lectura en modo asíncrono para evitar detener algunas tareas de las cuales también se encarga el thread. El ciclo de ejecución de cada thread incluye es capaz de detectar si hay una desconexión del extremo cliente del pipe, si esto sucede el thread espera la re conexión del extremo cliente del pipe para continuar con las operaciones de lectura.

El control de acceso a los buffers temporales donde son almacenados los datos de audio y video extraídos de los pipes se realiza a través de semáforos para evitar problemas de concurrencia al acceder a estos buffers desde otros objetos o procesos como se expone más adelante.

Los threads envían mensajes registrados en el sistema operativo en modo asíncrono para notificar periódicamente las acciones más importantes realizadas así como los posibles errores o excepciones que pudieran llegar a suceder.

Los mensajes de notificación se atienden en el hilo principal de ejecución del servicio generando mensajes en el archivo de bitácora donde se indica el problema reportado por los threads o si las condiciones son normales se agregan mensajes a la bitácora cada cierto tiempo en los que se indica que los threads no tienen ningún problema.

Envío de audio y video al hardware

En la sección anterior se describió la forma en la cual se diseñó la comunicación entre la aplicación Reproductor de Máster, encargada de la decodificación de los videos, y el servicio de interfaz con el software. Así mismo en la sección de control de hardware

también se describieron los pasos necesarios para iniciar la reproducción de la señal en una de las 4 tarjetas que incluye la tarjeta BlackMagic Decklink Quad.

Al inicio de la ejecución del servicio, en el thread de comandos incluyeron las operaciones para inicializar los dispositivos que están definidos en el archivo de configuración creando los objetos necesarios y preparando el hardware, de esta forma las tarjetas necesarias quedan listas para realizar las tareas de reproducción o captura según corresponda.

El diseño servicio de interfaz con el hardware incluyó la clase manejador de datos de salida. Esta clase sirve para crear un objeto encargado en tiempo de ejecución de decidir la fuente de la cual debe tomar los datos de audio y video para ser enviados a la salida, además realiza la combinación de los elementos que se deben incluir en la señal de salida dependiendo de los comandos que han sido ejecutados provenientes del Reproductor de Máster.

Al configurar el dispositivo de reproducción es necesario registrar una función de callback mediante la cual el hardware pueda notificar al servicio que un cuadro ha sido reproducido en la señal de salida, entonces el servicio envía un nuevo cuadro audio y video al buffer del dispositivo. El SDK del fabricante ofrece la posibilidad de registrar un objeto que contenga la función de callback para las notificaciones, en este caso se registra el manejador de datos de salida para que pueda responder a estas notificaciones.

Para la reproducción de material grabado almacenado en archivos se recibe el audio y el video provenientes de la aplicación Reproductor de Máster a través de los pipes y los threads dedicados se encargan de obtener la información para ingresarla a los buffers temporales. Si el manejador de datos de salida tiene registrado que la fuente de reproducción actual es un material grabado entonces obtiene el audio y video de los estos buffers temporales accediendo así a los datos provenientes del Reproductor de Máster y cuando se requiere los envía al hardware de salida.

Al iniciar la reproducción de una guía de programación si el primer evento a ejecutar es un material grabado el Reproductor de Máster empieza a decodificar el archivo y una vez que ha enviado 30 cuadros de video al servicio manda el comando de inicio de reproducción. El servicio recibe el comando de inicio de reproducción en el thread de comandos y ejecuta el pre-roll inicial de los cuadros de audio y video en el buffer del hardware, es decir se solicita al manejador de datos de salida que envíe 15 cuadros de audio y video provenientes de fuente grabada. El manejador obtiene la información de los buffers temporales de audio y video donde los threads dedicados a la lectura de los pipes almacenaron la información. Después de realizar el pre-roll inicial inmediatamente el servicio da la instrucción a la tarjeta de arrancar la reproducción de los datos que contiene el buffer del hardware, todo este proceso desde la recepción del comando de inicio de reproducción hasta iniciar la reproducción se realiza en el thread de comandos.

Una vez que el hardware ha entrado en estado de reproducción por cada cuadro que reproduce a la salida avisa al objeto manejador de datos de salida mediante el mecanismo de la llamada a la función de callback y el manejador de datos de salida se

encarga de enviar otro cuadro de video con el audio correspondiente al buffer del hardware manteniendo de esta forma un nivel estable en el buffer del hardware y por lo tanto la reproducción no se detiene. La clase manejador de datos de salida requiere realizar todas las operaciones necesarias para enviar un nuevo cuadro al buffer del dispositivo en un tiempo máximo de 33 milisegundos, ya que este es el tiempo requerido para desplegar un cuadro en pantalla, si la clase tarda más de ese tiempo en procesar el cuadro que debe enviar al buffer del hardware eventualmente dicho buffer se agotará y la reproducción por lo tanto se detendría.

El manejador de datos de salida fue diseñado y optimizado para realizar las funciones de combinación de sùpers, introducción de cuetones, normalización del nivel de audio, la combinación de la señal con el logo del canal y leer los cuadros de video y paquetes de audio de la fuente adecuada en un tiempo inferior al indicado, todas estas funciones serán detalladas más adelante.

Si desde la aplicación Reproductor de Máster se recibe un comando de fin de reproducción entonces el thread de comandos modifica las propiedades del objeto de manejo de datos de salida para que deje de enviar cuadros al buffer del hardware en respuesta a cada callback, además el thread envía la instrucción al hardware para detener la reproducción de forma inmediata, con lo cual se deja de generar la señal de salida.

Manejo de la señal de estudio

Otra característica que fue solicitada para el sistema de automatización es la posibilidad de incluir en la señal de salida una señal proveniente de estudios en vivo. Para realizar esta tarea en el servicio de interfaz con el hardware se incluyó la posibilidad de configurar un dispositivo dedicado a la captura de la señal de estudio, en el archivo de configuración del servicio se puede definir el dispositivo de la tarjeta Blackmagic Quad a utilizar.

Al iniciar la ejecución del thread de comandos se inicializa la tarjeta para la captura de la señal de estudio, cabe mencionar que esta tarjeta es opcional, en otras palabras si no se requiere puede indicarse en el archivo de configuración y no se inicializará ningún dispositivo para esta finalidad. Aunque el thread de comandos se encarga de inicializar el dispositivo para la captura no arranca en ese momento la captura de datos, es decir, se selecciona el dispositivo, configura el formato de captura, activa la tarjeta para captura de audio y video, y registra la función de callback solamente.

Antes de que el evento de señal de estudio, o también llamada señal externa, se ejecute el Reproductor de Máster, éste último envía un comando de pre-roll de señal de estudio al servicio. El comando de preroll de señal de estudio al ser ejecutado en el thread de comandos envía la instrucción al hardware de iniciar la captura de la señal mediante el dispositivo configurado anteriormente. El envío y ejecución de este comando de pre-roll sucede 3 segundos antes de que la señal de estudio se tenga que enviar a la salida, esto se debe a que el dispositivo no captura inmediatamente la señal de entrada en forma continua sino que tarda aproximadamente un segundo en estabilizarse y durante ese tiempo entrega cuadros de imagen intercalados con cuadros en negro, una vez que se

estabiliza el hardware la captura es ininterrumpida. Otra de las razones por la cual no se captura la señal todo el tiempo tiene que ver con la eficiencia del sistema, no utiliza dispositivos o recursos que no son necesarios en todo momento y por otro lado también se ha observado en la práctica que la ausencia prolongada de señal de entrada en las tarjetas de entrada puede ocasionar defectos en la entrega de la información capturada a las aplicaciones que interactúan con estas tarjetas como estar fuera de lipsync.

La captura en estos dispositivos, como ya se ha mencionado, requirió de una función de callback para entregar la información de la señal capturada cuadro a cuadro de video con su correspondiente paquete de audio, para este fin se creó en el servicio la clase manejador de entrada externa. En ejecución un objeto de esta clase contiene la función de callback y cada que recibe un cuadro capturado con su audio correspondiente, se valida la información de manera que cumpla con las características del formato de entrada, entonces si la validación es exitosa esta información es enviada al buffer temporal de la señal externa. El buffer de señal externa tiene la capacidad para almacenar hasta 2 segundos de señal capturada, es una cola circular de estructuras de datos que almacenan cuadros de señal externa, estas estructuras incluyen los campos necesarios para almacenar las imágenes en espacio de color UYVY y el paquete de audio correspondiente que puede incluir hasta 1602 muestras por canal.

Ya que la captura inicia con anterioridad en el momento en el cual debe ser transmitida a la señal de salida, el manejador de datos de salida descarta los cuadros del buffer de la señal externa a la velocidad del callback de la señal de salida con la finalidad de no saturar este buffer, y en la medida de lo posible no retrasar más la señal de estudio más de los 15 cuadros que constituyen el pre-roll inicial del hardware de salida. La condición para empezar a descartar cuadros de video es alcanzar un nivel inicial en el buffer temporal de señal externa de 10 cuadros, de esta manera el buffer mantiene un nivel estable y tiene un margen para enfrentar interrupciones que pudieran modificar el tiempo de respuesta de la llamada al callback de captura.

En el momento exacto que el reproductor de máster requiere que la señal externa sea enviada a la salida envía el comando de inicio de estudio, el cual es recibido en el thread de comandos y ejecutado. Al ser ejecutado se comunica al objeto manejador de datos de salida que debe introducir la señal externa en la señal de salida, es decir se modifica la fuente de los datos, de manera que en la siguiente notificación de callback de la señal de salida que recibe el objeto manejador de datos de salida ya no obtendrá el cuadro de audio y video de los buffers donde se almacena la información proveniente de los archivos grabados sino que toma la información de audio y video del buffer de señal externa y a partir de ese momento solamente toma los datos de ese buffer con lo cual se logra que la señal externa permanezca siendo transmitida a la salida. El envío de la señal externa a la salida continúa hasta que el reproductor de máster envíe un comando de fin de estudio, ese comando al ser recibido y ejecutado en el thread de comandos notifica al manejador de datos a la salida que debe dejar de tomar datos del buffer de señal externa y debe volver a una fuente grabada. Además con la ejecución de ese comando el thread de comandos ejecuta las funciones necesarias para detener la captura de la señal en el

hardware con lo cual se dejan de recibir llamadas a la función de callback de la señal externa.

Inserción de cuetones

En la transmisión de una señal de televisión existen las operaciones de inserción de comerciales o bloqueos, esto quiere decir que algunas cadenas de televisión aprovechan espacios dentro de los bloques de comerciales de la programación para insertar contenidos correspondientes a los anunciantes con los cuales tienen contratos comerciales. La inserción de comerciales puede ser automática o manual, aunque actualmente en la televisión como en otras industrias se prefieren los procesos automáticos por la relación costo-beneficio que representan. Durante la operación de inserción el contenido de la señal original es reemplazado por el contenido propio que se va a insertar, por eso también se denominan bloqueos a estas operaciones.

Generalmente esta operación solamente se realiza en espacios autorizados e identificados para este fin por los sistemas de automatización, los espacios a su vez fueron designados por algún usuario que haya planeado la guía de programación y conoce cuáles contenidos pueden ser sujetos de operaciones de bloqueo, el resto de los comerciales que no son autorizados para bloqueo se debe a que los anunciantes pagan a la cadena que genera la señal del canal para que su comercial esté en pantalla en todos los lugares donde la señal sea emitida y por lo tanto no se pueden eliminar de la señal.

En la actualidad existen varios métodos para incluir la señalización de los bloques comerciales en los cuales pueden realizar la inserción de contenidos las demás televisoras que retransmiten la señal, algunos mecanismos de señalización son digitales y otros aún son análogos. Los mecanismos de señalización incluyen alguna forma para indicar el momento exacto en el cual se puede iniciar la inserción de contenidos sobre la señal y una forma de avisar que se ha terminado el espacio asignado para los bloqueos. En particular hay uno de muy amplio uso en las televisoras alrededor del mundo que se llama "cuetone".

Un cuetone es un mensaje compuesto por tonos de audio que se utiliza para ejecutar una acción. Dentro de la transmisión de televisión los cuetones son tonos DTFM (**Dual-tone multi-frequency signaling**) los cuales se escuchan como los tonos que existen en las líneas telefónicas donde de hecho tuvieron su origen. El mecanismo de DTFM fue originalmente utilizado por Bell System en Estados Unidos, está estandarizado bajo la recomendación ITU-T Q23. La DTFM tiene tonos para representar los 10 dígitos del sistema decimal así como letras de la A a la D y dos caracteres especiales # y *.

Los tonos se generan en base a la suma señales senoidales, una de estas señales tiene una frecuencia baja que está entre 697 y 941 Hz y la otra tiene una frecuencia denominada alta que puede ir de los 1209 Hz a los 1633 Hz, fueron diseñados para ser recuperados en el otro extremo del canal de comunicación sin ningún problema. Debido a

estas dos frecuencias es que se denominan tonos duales y dado que cada frecuencia maneja está dentro de un rango distinto reciben también la denominación multifrecuencia.

La identificación de los tonos, que también he realizado en otros proyectos, se basa en el análisis de la transformada rápida discreta de Fourier de la señal recibida cuya implementación en los equipos actuales de cómputo está optimizada y su ejecución es muy rápida. Una vez que se calcula la transformada rápida discreta de Fourier se revisa si están presentes las componentes de las frecuencias centrales de los tonos y de esa manera es posible reconocer los dígitos o caracteres que se enviaron así como el orden en el cual han llegado.

Los cuetones son un mecanismo de señalización análogo aunque su generación y recuperación puede realizarse en sistemas digitales. Existen otros mecanismos como el SCT-35 que fue creado para la señalización de señales digitales, este tipo de señalización generalmente se ocupa en la transmisión de video a través de redes de datos.

En el caso de este sistema de automatización fue solicitado únicamente el soporte para cuetones, el cual era el de uso más extendido en los mercados a los cuales llegaría la señal generada en los canales para los cuales fue diseñado el sistema. Los tonos de audio se generaron mediante el cálculo de la suma de funciones senoidales a partir de expresiones matemáticas con las frecuencias antes descritas, para cada dígito se tiene un par de frecuencias diferentes. Los resultados de las funciones son muestreados a 48 kHz, para cada tono se creó un buffer de datos en el que cada muestra calculada es guardada tomando en consideración el orden que los datos a nivel de bytes que requieren para ser enviados como audio a la tarjeta BlackMagic Decklink Quad.

Dentro de la industria de la televisión existen dos secuencias estándar de tonos, una para indicar el inicio de un segmento de comerciales autorizado para inserción y otra para indicar el final del segmento de comerciales donde se autoriza la inserción de comerciales. La secuencia de tonos que marca el inicio de la autorización de operaciones de bloqueo es 123* y la secuencia de fin del bloque autorizado es 123#.

Como parte de los requerimientos del sistema de automatización fue solicitado que los cuetones se insertaran siempre en los canales de audio 7 y 8 de la señal generada, es por ello que el formato de audio en el hardware de salida siempre se inicializa a 8 canales. Otra regla solicitada fue que debían ser incluidos en la señal 3 segundos antes de que inicie o finalice el bloque comercial donde se autoriza la inserción de comerciales a las televisoras que retransmiten la señal. La inserción de los cuetones en los canales de audio 7 y 8 tiene su justificación debido a que esos canales generalmente no son transmitidos al aire para el usuario final sino que solamente se ocupan en los sistemas intermedios. Dependiendo del uso que se le dan a los canales de audio en todos los equipos donde se procesa la señal es que se decide en este tipo de sistemas de automatización en qué canal de audio deben insertarse los cuetones, esos canales no deben ser audibles para el televidente.

En el sistema de automatización desarrollado la aplicación Reproductor de Máster que lleva a cabo el control de la guía envía un comando al servicio de interfaz con el hardware en el momento preciso en el que se necesita realizar la inserción de la secuencia de cuetone ya sea de inicio o fin de bloque, por lo tanto existe un comando para indicar que debe realizarse la inserción de cuetone de inicio y otro para indicar que se debe insertar el cuetone de fin. Cabe mencionar que no todo el tiempo hay en la señal de salida cuetones, solamente se activan para señalar el inicio o final de los segmentos de comerciales bloqueables. El comando viaja a través del pipe de comandos, es recibido por el thread de comandos, el cual se encarga de su ejecución modificando una bandera dentro del objeto manejador de datos de salida. Entonces a partir de la siguiente llamada de callback de la tarjeta de salida el manejador de datos de salida incluirá en el buffer de audio que envía al hardware de salida en las posiciones de los bytes correspondientes a las muestras de los canales 7 y 8 la información de las muestras del cuetone necesario, es por esto que el conocer la forma en la cual se interpretan los datos de las muestras y su posición dentro los bytes de la información enviada al hardware es imprescindible en este tipo de proyectos. El manejador de datos de salida controla la inserción de la secuencia del cuetone de principio a fin registrando el conteo del número de muestras que ha introducido en los paquetes de audio enviados a la salida hasta que se agotan en el buffer de muestras del cuetone. Los valores de las muestras que se almacenan en los buffers correspondientes a cada secuencia de cuetone se calculan al crear la instancia del objeto manejador de datos de salida.

Inserción de logo del canal en la señal

En la industria de la televisión cada canal emitido cuenta con una estrategia de promoción e imagen como cualquier otro producto comercial, el tipo de promoción e imagen que tiene un canal busca llamar la atención de los televidentes con el objetivo de captar audiencia.

Las estrategias de imagen de un canal de televisión cuentan con varios elementos en pantalla como son:

- Cortinillas: Son los gráficos que aparecen como parte del bloque de programa en la salida a corte comercial o bien al regreso del corte comercial. Generalmente están acorde a la imagen del canal o pueden ser parte de la promoción e imagen de la producción televisiva en la que se incluyen. Ejemplo de este tipo de gráficos son los anuncios de en un momento regresamos o continuamos con.
- Promocionales: Materiales grabados en video que forman parte del corte comercial, sin embargo a diferencia de los comerciales contratados por los anunciantes contienen información acerca de los programas que se transmiten en el canal, avisos de estrenos, avisos de eventos especiales, cambios en la programación o avances de ciertos programas con los cuales se busca cautivar a la audiencia.
- Logo del canal: Es un gráfico que se agrega generalmente sobre los bloques de programa del canal en alguna de las 4 esquinas de la pantalla. Representa la

imagen y personalidad del canal. Este gráfico está presente durante toda la duración del bloque de programa y puede ser estático o con una animación que se repite cada cierto tiempo. Cuando el programa se va a corte comercial el logo desaparece de la pantalla.

- Súpers o branding: Son gráficos que también se sobreponen en la pantalla acorde a la imagen del canal y pueden anunciar horarios de otros programas, eventos especiales, información acerca del canal, indicar el nombre del programa que se está transmitiendo en ese momento, la clasificación para la audiencia del programa o cualquier otra información relevante.

El sistema de automatización desarrollado puede manejar todos estos elementos para la promoción del canal. Las cortinillas suelen venir como parte de la edición de los bloques de programa o bien pueden ser videos ingestados con este contenido que se colocan al principio o al final de los cortes comerciales en la guía de programación. Los promocionales se envían a través del módulo de ingestión destinado para comerciales y son incluidos en la guía de programación dentro de los cortes comerciales.

La inserción del logo del canal en la señal de salida también fue soportada, la combinación con la imagen de fondo se realiza en el servicio de interfaz con el hardware. Se tienen las siguientes opciones para incluir el logo en la señal:

- Logo estático
- Logo animado
- Gráfico animado acompañado de gráfico estático

Logo estático

El manejo de logo estático se planteó de manera que los diseñadores entregaran una imagen a resolución completa por cada tipo de señal de salida que tenga el módulo de reproducción. Si el módulo de reproducción solamente tiene señal de salida en formato HD el diseñador debe entregar una imagen con resolución 1920x1080 en espacio de color RGBA en cualquier formato de archivo que permita incluir el canal alpha, generalmente es PNG o TGA. En caso de que el módulo de reproducción solamente tenga señal de salida en formato SD entonces se entrega una imagen con resolución 720x480 en espacio de color RGBA. También existe la posibilidad de que el máster esté configurado para generar señal de salida tanto en formato HD como en formato SD a través de dos dispositivos diferentes, lo cual se define en la configuración de servicio, en ese caso el diseñador debe entregar las imágenes en ambas resoluciones.

El soporte de esta característica basado en la entrega de la imagen del logo a resolución completa se diseñó de esta manera para que el diseñador gráfico colocara el logo en las coordenadas exactas que necesita dentro de la pantalla. Existen varias plantillas y medidas sobre la pantalla que los diseñadores conocen, las cuales deben ser respetadas para la correcta transmisión y posición del logo a pesar que la señal pase por diversos procesos de conversión.

Algunas de las señales por ejemplo serían emitidas solamente en formato HD desde el sistema de automatización y transmitidas a través de distintos equipos y sistemas, hasta ser recibidas en los diferentes países donde serían retransmitidas en los cuales si solamente contaban con la infraestructura para transmisión en formato SD eran sometidas a un proceso de transformación en el cual pasan de formato HD a SD, dicha transformación se conoce como down-conversion. Existen varios métodos para realizar down-conversion, uno de ellos consiste en realizar un corte de la porción central de la pantalla a manera de obtener una porción de la pantalla con proporción 4:3 a diferencia de la proporción 16:9 del formato HD, este tipo de conversión se llama Central Cut.

En la transformación antes descrita si el logo está fuera del área que incluye el corte entonces sería eliminado por este proceso, para asegurarse de que esto no suceda los gráficos y logos se posicionan en la pantalla de manera que queden dentro al área que incluye el recorte y es conocida como safety 4:3.

Las imágenes que entregan los diseñadores con el logo se definió que serían en el arranque del servicio, el objeto manejador de datos de salida se encarga de esta tarea. En esta clase se integró la librería para manejo de imágenes DevIL (Developer's Image Library) que cuenta con las funciones suficientes para la lectura de una gran cantidad de formatos de archivos de imágenes entre los cuales están los archivos en PNG y TGA. Una vez cargado el gráfico se transforma con el proceso descrito en el thread de placas para llevarlo al formato UYVY con canal alpha y es almacenado en una estructura diseñada para guardar este formato.

El control de la inserción de logos en la señal de salida generada por el servicio se realiza a través de los comandos activar logo y desactivar logo enviados por el Reproductor de Máster. El Reproductor de Máster tiene el registro del momento en el cual debe aparecer el logo en pantalla y desde esa aplicación se envían los comandos al servicio, los cuales permiten activar la inserción del logo en pantalla o desactivarla. La aplicación Reproductor de Máster envía el comando activar logo unos segundos después de iniciar un bloque de programa y el comando desactivar logo se envía unos segundos antes de terminar un bloque de programa, además tiene un control manual que le permite a un operador activar o desactivar la inserción del logo manualmente.

Los comandos para el manejo del logo viajan a través del pipe de comandos y son leídos en el thread de comandos, a su vez este thread se comunica mediante mensajes con el objeto manejador de datos de salida, este último objeto al recibir las notificaciones activa o desactiva una bandera propia del objeto que indica la si la inserción del logo está activa.

La combinación del gráfico del logo con cada cuadro de video que el objeto manejador de datos de salida enviará al hardware de salida se realiza en tiempo real sin importar la fuente del cuadro siempre y cuando la bandera de inserción de logo está activada. La combinación se lleva a cabo dentro del objeto manejador de datos de salida con la siguiente serie de pasos:

1. Verificar la bandera de logo activo, si está activa se aplican los siguientes pasos, caso contrario no se aplican.
2. La imagen del logo convertida a espacio UYVY con canal alpha se recorre revisando primero el arreglo de banderas de canal alpha por renglón para optimizar el proceso, si un renglón no tiene canal alpha entonces salta al siguiente.
3. Si el renglón tiene canal alfa se procede a revisar el valor del canal alpha en cada píxel del renglón.
4. Si el valor de canal alpha del pixel es 0 se pasa al siguiente pixel, de esta forma el valor de las componentes del pixel con alpha channel 0 se mantiene como se encuentra en el cuadro de video original, es decir en la señal de fondo.
5. Si el valor del canal alpha para ese pixel es de 200 o superior entonces ese pixel del cuadro de la señal de fondo toma el valor del pixel correspondiente en el gráfico del logo ya que no vale la pena realizar los cálculos, no hay diferencia apreciable a la salida.
6. Si el valor del canal alpha es mayor a 0 y menor a 200 se utiliza una expresión de interpolación entre la imagen del cuadro de fondo y el gráfico del logo en todas sus componentes. Este proceso afecta dos píxeles por la naturaleza del espacio de color UYVY.
7. Se recorre de esta manera toda la imagen del gráfico del logo hasta llegar al último renglón, por lo tanto después de recorrer los pixeles de un renglón el proceso regresa al paso 2.

En el proceso descrito se observa la importancia de los arreglos de banderas por renglón definidos en el formato UYVY con canal alpha, permiten optimizar y realizar de forma más rápida la combinación descartando renglones enteros de la imagen en los cuales no es necesario revisar ni hacer ningún cálculo.

La expresión utilizada para la combinación con la imagen de fondo es la siguiente:

$$V_p = [((v_{ps} - v_{pf}) * v_{ca}) / 255] + v_{pf}$$

Donde:

V_p es el valor nuevo del pixel

v_{ps} es el valor del pixel de la imagen sobrepuesta

v_{pf} es el valor del pixel de la imagen fondo

v_{ca} es el valor del canal alpha para ese pixel en la imagen del logo

Esta expresión se encuentra generalmente en textos como alpha compositing, la cual ha sido simplificada ya que los valores de los pixeles solamente incluyen valores posibles de 0 a 255, por otro lado la expresión original puede incluir un valor del canal alpha para la imagen de fondo sin embargo en este caso toma el valor máximo de 255.

Una vez que se termina de recorrer toda la imagen calculando los valores nuevos de los pixeles, en los cuales es necesario, el logo queda combinado con la imagen de fondo, que puede provenir de una fuente grabada o de señal externa, la imagen resultante es enviada a la salida. Si el módulo de salida cuenta con dos formatos de salida, tanto en HD como SD, el logo se combina por separado con la imagen de fondo que se enviará a la salida en formato HD y por otro lado en la imagen en SD que se enviará a la salida SD. El manejo de la conversión de formato para lograr tener ambas salidas en simultáneo se expondrá más adelante.

Logo animado

El soporte para logo animado fue un requerimiento que surgió una vez que el sistema ya estaba en operación, la petición fue soportar logos con animaciones. El desarrollo a la medida de este sistema de automatización permitió agregar nuevas características con un bajo costo de desarrollo a diferencia de una solución comercial en la cual cada nuevo requerimiento requiere una gran inversión tanto monetaria como en de espera.

El soporte del logo animado se planteó a través del manejo de una secuencia de imágenes que componen la animación. A diferencia del logo estático la secuencia de imágenes solamente contiene un área rectangular que encierra con muy poco margen el gráfico del logo, todas las imágenes deben tener el mismo tamaño encerrando el gráfico de mayor tamaño presente en la animación. Se definió que las imágenes de las secuencias deberían ser entregadas en resolución HD, o en resolución SD o en ambas resoluciones dependiendo de los formatos de salidas configurados en el servicio de interfaz con el hardware.

Los nombres de los archivos de imágenes deben ser todos iguales y terminar con un índice generalmente a 3 o 4 dígitos antes de la extensión del archivo. Se definió una subcarpeta dentro de la ubicación donde se encuentra instalado el servicio con el nombre de logo_animado donde se debe colocar la secuencia en resolución HD y para la salida en el formato SD la subcarpeta se llama logo_animado_SD. El tipo de logo que incluirá el servicio se define en el archivo de configuración, se puede configurar para insertar el logo estático o animado y son mutuamente excluyentes.

Cada imagen de la secuencia que constituye el logo animado se carga al arrancar el servicio en el objeto manejador de datos de salida a través de la integración de la librería DevIL, se transforma al espacio de color UYVY con canal alpha y es almacenado en un arreglo de estructuras. La carga de los archivos de la secuencia de imágenes del logo se hace listando todos los archivos de los subdirectorios antes mencionados dependiendo de la configuración del formato de la señales de salida, debido al listado es que los nombres deben incluir el índice para conservar el orden de las imágenes que componen la animación diseñada en la señal de salida. La animación de este tipo de logos generalmente dura de 10 a 20 segundos, tomando en cuenta que el framerate de la señal de televisión es de 29.97 frames por segundo nos da aproximadamente entre 300 y 600 imágenes para esas duraciones. Debido a la cantidad de imágenes que eran necesarias

es que los logos animados solamente deben incluir el área efectiva con gráficos para minimizar el uso de memoria al almacenar las imágenes.

Con la secuencia de imágenes almacenada en memoria el proceso para combinarla con la señal es prácticamente el mismo que con el logo estático, también se utilizan los comandos de activar y desactivar logo que son enviados desde el reproductor de máster y en cuanto son recibidos por el thread de comandos actualiza la bandera en el objeto manejador de datos de salida para que inserte o no el logo con la secuencia de pasos antes descrita. Las diferencias con el logo estático son 3:

- Ya que cada imagen de la secuencia sólo contiene el área del gráfico y no la pantalla completa, entonces fue necesario que se entregara como parte de la información del logo la coordenada superior izquierda donde será colocado. La coordenada en se incluye en el archivo de configuración del servicio en pixeles y la información se carga al arrancar el servicio.
- Por otro lado debido a que las imágenes solamente contienen el área del gráfico no se recorre el cuadro de fondo a resolución completa en el proceso sino solamente el área rectangular incluida en las imágenes a partir de la coordenada donde se posiciona el logo, salvo esta diferencia la secuencia de pasos es la misma.
- Al ser una animación el objeto manejador de datos de salida lleva el registro del índice de la imagen del logo que inserta en cada de callback de salida, de esta manera va recorriendo la secuencia de imágenes que componen la animación. En cuanto detecta que ha llegado al último cuadro de la animación se queda combinando ese cuadro con la señal el tiempo que se especifique en el archivo de configuración del servicio, en otras palabras realiza una pausa en la animación manteniendo en pantalla el último cuadro de la secuencia de imágenes. Una vez que se supera el tiempo definido para la pausa reinicia el índice a 0 con lo cual se reinicia la animación y vuelve a avanzar en la secuencia de imágenes logrando así en pantalla tener la animación en un ciclo que se repite una y otra vez junto con la pausa correspondiente hasta que no llegue un comando de desactivar logo.

El soporte para el logo animado hace que la imagen del canal sea aún más sofisticada y resalte a la vista del televidente a diferencia del logo estático. Si el módulo de reproducción está configurado para entregar señal tanto en formato HD como en SD la coordenada es diferente en las dos resoluciones por lo que debe ser especificada para configurar el servicio de interfaz con el hardware.

Gráfico animado acompañado de gráfico estático

Un requerimiento solicitado posteriormente al soporte para logo animado fue la posibilidad de poder acompañar el logo animado con un gráfico complementario estático por ejemplo en otra de las esquinas de la pantalla. Este mecanismo se puede ocupar en alguna campaña de imagen donde además del logo animado es necesario incluir el anuncio de algún estreno importante o puede aprovecharse de forma contraria al colocar un logo estático y una animación como propaganda por ejemplo del aniversario del canal.

Ya que fue el último requerimiento en el caso del soporte para logos en ser desarrollado se hizo con toda la flexibilidad posible con base en el conocimiento y experiencia adquiridos en el soporte de los dos tipos de logo que se han expuesto.

Para el gráfico animado se utilizó la misma implementación que la descrita en la sección anterior de logo animado y tiene las mismas especificaciones. En cuanto al gráfico estático la implementación es similar a la descrita en el apartado de logo estático, la única diferencia es que se dio la libertad de entregar la imagen en la resolución completa o bien en cualquier otro tamaño que incluya solamente al gráfico a combinar con la señal, si la resolución es menor a la de pantalla completa debe entonces acompañarse de la coordenada superior izquierda donde se desea colocar el gráfico estático sobre la pantalla de la señal de salida.

La operación es similar internamente a los otros logos salvo las siguientes diferencias:

- Al arrancar el servicio verifica la configuración cargando tanto la secuencia de imágenes para la animación en los formatos necesarios así como el gráfico estático y las coordenadas para ambos elementos en los formatos de salida.
- Activar y desactivar la combinación del logo con la señal se realiza a través del envío de comandos activa logo y desactiva logo desde el reproductor máster. El comando activa o desactiva la inserción ambos elementos.
- Si el logo está activo, primero se combina el cuadro correspondiente de la secuencia animada con la señal y posteriormente el gráfico estático, en ambos casos solamente se recorren los pixeles presentes en el área de los gráficos a superponer y además está optimizado con el arreglo de banderas de canal alpha por renglón.

En caso de utilizar esta alternativa para incluir los logos, se definió que el diseñador debe entregar tanto la secuencia de imágenes como el gráfico estático en las resoluciones configuradas en el servicio de interfaz con el hardware. La carga de los archivos de imágenes se apoya en la integración de la librería DevIL y el proceso de conversión de imágenes a formato UYVY con canal alpha descrito en la sección del thread de plecas.

Inserción de súpers en la señal de salida

Un contenido de tipo súper es una secuencia de imágenes que componen una animación y los gráficos cubren solo una porción de la pantalla para incluir diversos anuncios superpuestos en la señal de salida.

En la sección de ingestión se explicaron las características de los gráficos que se entregan al sistema para convertirlos en archivos de video en formato MOV sin audio, por otro lado en la sección del thread de plecas se expuso cómo es que el servicio recibe los cuadros decodificados de estos materiales en formato MOV desde el Reproductor de Máster y cómo estos son convertidos a espacio de color UYVY con canal alpha, además de ser almacenados en un buffer temporal dedicado para guardar estos cuadros de video en espera de ser insertados en la señal de salida.

El inicio de reproducción de un súper se sincronizó a la cantidad de cuadros que se han enviado al buffer temporal, el thread de plecas en cuanto detecta que ha enviado exitosamente 10 cuadros al buffer temporal envía un mensaje del sistema operativo dirigido al objeto manejador de datos de salida, el cual constituye una notificación que le avisa al objeto que el súper está listo para ser incluido en la señal de salida. Al recibir la notificación el manejador de datos de salida enciende la bandera de inserción de súper. A partir de la siguiente llamada de callback del hardware de salida, si la bandera de inserción de súper está activa, en cada llamada de callback se extrae un cuadro del buffer temporal de súper para ser combinado con la imagen que será enviada a la salida del sistema.

Los súper eran entregados a resolución de pantalla completa, si el sistema de automatización estuviera configurado para entregar señal de salida en dos formatos diferentes entonces se definió que se entregarían en la resolución más alta y los gráficos deberían estar ajustados dentro del área safety 4:3 para su correcta visualización en ambos formatos, como se ha mencionado el proceso para generar las dos señales en simultáneo se expone más adelante.

El proceso de combinación de cada cuadro de un súper con la imagen de fondo es idéntico al proceso de combinación de un logo estático, es por ello que los cálculos del proceso han sido optimizados incluso con operaciones de corrimientos de bits y tomando en cuenta el umbral del valor del canal alpha que no representa una diferencia visible a la salida para evitar hacer más cálculos.

La diferencia es que el cuadro del súper se va obteniendo del buffer temporal donde el thread de las plecas los almacena y el objeto manejador de datos de salida detecta el momento en el cual se han terminado los cuadros de ese buffer, en ese momento se apaga la bandera de combinación de súper y no se obtienen más cuadros del temporal buffer hasta que se reciba otro mensaje notificando acerca de un nuevo súper listo para ser combinado con la señal de fondo.

En una sección posterior se expondrá el orden en que se realizan todas las operaciones soportadas por el servicio en forma de un resumen donde quede claro en qué momento se ejecuta cada operación.

Manejo de salida en formato HD y SD simultáneo

El sistema de automatización tuvo como otro requerimiento la posibilidad de contar con señales de salida tanto en resolución HD como en SD de manera simultánea, esto ya ha sido mencionado en varias ocasiones dentro del informe sin embargo hasta este punto no se ha detallado cómo se realiza esta tarea.

El manejo de la salida en ambas resoluciones se lleva a cabo en el servicio de interfaz con el hardware, la configuración de las señales de salida en el servicio se diseñó con soporte para las siguientes opciones:

- Salida sólo en formato HD

- Salida sólo en formato SD
- Salida en formato HD y SD simultáneamente

En el caso de necesitar solamente un formato de salida tanto el Reproductor de Máster como el servicio de interfaz con el software se configuran para soportar sola una tarjeta de salida y además en el formato que sea necesario ya sea HD o SD. Para configurar el modo con salida en formato HD y SD simultáneos el Reproductor de Máster se configura para decodificar los videos en formato HD y el servicio de interfaz con el software se configura con dos dispositivos de salida y una bandera que indica que el primer dispositivo configurado entregará la salida en formato HD y el segundo en formato SD. Si la bandera de formato HD y SD no es activada entonces el servicio tendrá dos salidas con el mismo formato en los dispositivos seleccionados, esta característica se diseñó para extender más las capacidades del sistema de automatización. El formato de captura de la señal externa se configura en el servicio de forma automática de manera que concuerda con el formato de la salida o bien cuando está configurado para salida en simultáneo en formato HD y SD internamente la tarjeta de entrada es configurada para capturar video en formato HD.

Existen métodos de down-conversion en los cuales se transforma una imagen de formato HD a SD y también existen los métodos de transformación up-conversion con los cuales se pasa de formato SD a HD. En las transformaciones down-conversion se descarta información de una u otra manera de la señal original en formato HD porque el formato SD tiene una cantidad mucho menor de píxeles, por otro lado en los métodos de up-conversion, en los cuales se pasa de un formato con menor número de píxeles a uno con mayor cantidad de píxeles, los valores de los píxeles faltantes para generar un cuadro de mayor resolución deben ser interpolados de alguna manera para crear los píxeles necesarios además de pasar de una relación de aspecto 4:3 a una relación de aspecto 16:9, todo lo anterior nos lleva a perder calidad en la imagen o tener deformaciones de las figuras cuando se hace un proceso de up-conversion en vez de recibir la señal de manera nativa en HD desde el principio. En las transformaciones de down-conversion dado que se descarta información del formato HD no se tienen ningún impacto en términos de pérdida de calidad de imagen en el formato resultante SD. Es por esto que cuando se requiere salida en simultáneo en formato HD y SD se decidió que tanto la decodificación de videos como la captura de la señal externa se realizaran en formato HD.

El procesamiento del video para generar la señal de salida dentro del servicio se puede resumir en esta serie de pasos:

1. El manejador de datos de salida obtiene el cuadro de video a enviar a la salida dependiendo de la fuente, es decir, de los buffers de los materiales grabados o del buffer de la señal de estudio externa.
2. Sobreponer el súper si es necesario al cuadro obtenido en el paso anterior.
3. Si está configurado para formato de salida en HD y SD simultáneo se hace la operación de down-conversion central cut y el resultado es almacenado en un buffer diferente, caso contrario en el cual la salida es solo en un formato pasa al paso 6.

4. Con el cuadro resultante de la operación central cut si la bandera de sobreponer el logo está activa se combina con los gráficos necesarios correspondientes al logo en formato SD.
5. El cuadro con central cut procesado en el paso anterior se envía a la tarjeta de salida configurada para entregar la señal en formato en SD.
6. Sobreponer el logo si está activa la bandera correspondiente en la imagen resultante del paso 2. Si está configurado en modo HD y SD se sobrepone siempre el logo o gráficos correspondientes al logo en HD.
7. Enviar el cuadro de video a la tarjeta salida correspondiente.

El proceso descrito en la serie de pasos anterior se optimizó para ser realizado en menos de 33 milisegundos por cada cuadro a enviar a la señal de salida, de otra forma tendríamos una condición de sobreflujo y la reproducción de la señal de salida tendrá defectos. Todo el proceso se realiza en el objeto manejador de datos de salida. Además del procesamiento de video el servicio de interfaz con el hardware lleva a cabo el procesamiento de audio simultáneamente como se detalla más adelante.

La transformación para convertir la imagen de formato HD a SD aplicando la operación de central cut se realiza de la siguiente manera:

- Como el nombre de la transformación lo indica se debe realizar un corte de la porción central del cuadro en formato HD a fin de extraer una zona de la imagen en proporción 4:3. Esto se hace tomando como base el tamaño de alto de la imagen en HD que es de 1080 pixeles, dado que la relación de aspecto en formato SD es 4:3 si calculamos el número de pixeles a lo ancho que le corresponden a ese alto resultan ser 1440 pixeles. Entonces el área a recortar mide 1440 x 1080 pixeles, a lo ancho del cuadro se requiere descartar la diferencia entre 1920 y 1440 que son 480 pixeles, pero se reparten en ambos lados para hacer un corte central, es decir 240 pixeles de cada lado son descartados en el proceso de recorte. En el caso del servicio el recorte se hace en el objeto de manejo de datos de salida copiando las porciones del buffer de la imagen en HD correspondientes a al área de recorte en cada renglón y descartando los bytes correspondientes a la información fuera del margen.
- El resultado del paso anterior es la imagen recortada en resolución 1440 x 1080 pixeles, esa imagen se escala a la resolución 720 x 480 mediante la integración de la librería swscale que forma parte del conjunto de librerías libav.
- El resultado es una imagen con resolución en 720 x 480 donde se han descartado las partes laterales de la imagen en formato HD y está en espacio de color UYVY lista para ser enviada a la tarjeta configurada en formato SD.

Normalización automática del nivel de audio

La señal de salida no solamente está compuesta por video o imagen sino también cuenta con audio. El audio se extrae de la misma fuente que el cuadro de video a enviar a la salida, si el video proviene del buffer de video decodificado entonces el paquete de audio a enviar a la salida se extrae del buffer de audio decodificado, pero si el video es extraído

del buffer de la señal externa entonces el audio se extrae de manera simultánea del mismo buffer ya que la estructura para almacenar la señal externa contiene ambos elementos.

Las operaciones efectuadas al video no afectan el audio, el cual es procesado para insertar los cuetones en la señal, como se expuso en una sección anterior.

El procesamiento de audio se puede resumir de la siguiente manera:

1. Extraer el audio del buffer de audio decodificado o del buffer de señal externa dependiendo de la fuente del video que se va a enviar a la salida.
2. Normalización del nivel de audio
3. Introducción del cuetone si es necesario
4. Enviar el audio a los dispositivos de salida, si se tiene salida en dos formatos de video primero se envía al dispositivo en SD y luego al dispositivo en formato HD como se hace con el video. En caso de un solo formato de salida no importa a qué tarjeta se mande en caso de tener dos.

Los pasos 1, 3 y 4 ya han sido expuestos en este informe, ahora es momento de explicar el paso 2 de este proceso.

Los materiales que conforman la guía podían ser enviados desde diferentes áreas de edición, algunos podían provenir incluso de editores externos como es el caso de los anunciantes cuyos contenidos son generados en alguna agencia publicitaria. Los programas generalmente se generaban siempre en las mismas cabinas de edición pero los promocionales se enviarían de cualquier lugar que contara con los privilegios de acceso necesarios para utilizar el subsistema de ingestión Web. La guía de programación podía contener también bloques de programa en vivo cuyos niveles de audio podrían llegar a ser muy diferentes a los materiales grabados.

En el caso del video los parámetros tales como la resolución y el framerate están estandarizados, para el audio también se puede estandarizar el número de canales y la frecuencia de muestreo de la señal sin embargo en cuanto al nivel de audio o volumen era sumamente difícil lograr que todas las áreas involucradas en la generación de contenido o bien en la transmisión de señal en vivo manejaran el mismo nivel de audio. Una muestra de esto se puede encontrar en cualquier señal de televisión abierta o sistema de televisión de pago, al salir de bloque de programa a cortes comerciales muchas veces se escuchan más altos o más bajos los comerciales, lo hace que el televidente en ocasiones incluso tenga que hacer un ajuste en el volumen de su televisor.

Desde hace algunos años se habían realizado esfuerzos, investigaciones y desarrollo dentro del área para contar con un método de normalización de audio eficaz y que permitiera mantener el volumen de la señal de audio no importando si se está en vivo, es un programa grabado o bien la programación está en corte comercial para mejorar así la experiencia del televidente.

Una de las dificultades para lograr un volumen estable tiene que ver primero con la unidad de medición a elegir, en el ámbito del procesamiento de audio digital se cuenta con la unidad de medida dBFS.

La abreviatura dBFS significa “decibelos a escala completa” (decibels full scale). Es una unidad de medida para los niveles de amplitud de una señal en decibelios dentro de sistemas digitales que manejan una amplitud máxima.

En esta escala el máximo valor posible es 0 dBFS. La forma de calcularlo está dada por la siguiente expresión:

$$\text{dBFS} = 20 * \log_{10} (\text{valor_muestra} / \text{maximo_valor_amplitud})$$

En el sistema de automatización hay que recordar que los valores de cada muestra de audio son almacenados en variables de 2 bytes con signo, por lo tanto el valor máximo de amplitud posible es 32767, el valor de las muestras en estos cálculos deben trabajarse en valor absoluto.

El cálculo se ajusta exactamente a la descripción ya que si sustituimos el valor máximo posible en el valor de la muestra nos queda lo siguiente:

$$\text{dBFS} = 20 * \log_{10} (32767 / 32767) = 20 * \log_{10}(1) = 20 * 0 = 0$$

Por otro lado las muestras con valores de amplitud muy pequeños generan valores más y más negativos en la escala dBFS, aunque la escala tiene un límite ya que solo es posible poder calcular el correspondiente valor en dBFS de hasta una muestra con valor 1, porque para el valor de 0 no es posible determinarlo debido a la naturaleza de la función logarítmica. De manera experimental cualquier valor por debajo de -60 dBFS no es ni siquiera percibido por el oído humano, de hecho corresponde a ruido presente siempre en los componentes electrónicos involucrados en la transmisión de la señal.

Con la escala adecuada vale la pena preguntarse cómo se debe llevar a cabo la medición en el terreno digital del audio, ya que una medición muestra a muestra poco nos puede indicar acerca del volumen en general que tiene el audio para la percepción humana. Lo que se hace entonces es trabajar a nivel de paquetes de muestras correspondientes en este caso a cuadros de video, como ya se ha comentado los paquetes contienen entre 1600 y 1602 muestras. Obtener el nivel de un paquete de audio se hace de la siguiente manera:

1. Obtener el valor promedio de los valores absolutos muestras de todos los canales del paquete.
2. El promedio se divide entre la raíz cuadrada de 2 para obtener su valor RMS.
3. Verificar que el resultado de la operación anterior no exceda el valor máximo por algún error ni tampoco sea menor o igual a cero, si alguno de los dos casos sucede se ajusta al valor máximo o a un mínimo de 1.
4. Aplicar la expresión para obtener el valor correspondiente en escala dBFS.

5. Finalmente si el valor resulta ser menor a -60 dBFS se asigna -60 dBFS debido a que es un nivel no audible e incluso no es común que se dé un resultado con ese valor final.

El proceso requiere trabajar con valores RMS o eficaces ya que es ideal para medir la intensidad en señales con formas de onda periódicas, para efectos prácticos en ondas sinusoidales se obtiene dividiendo el valor de la amplitud de la onda entre la raíz cuadrada de 2. El valor RMS se ajusta mejor al modelo de percepción humana que es lo que buscamos en este método de normalización, es decir, realizar la normalización con base en la percepción del televidente de manera que el nivel de audio permanezca muy similar de un contenido a otro.

En la sección dedicada al thread de audio se mencionó que para cada paquete de audio proveniente de material grabado se obtenía el valor RMS del nivel de audio, esto se hace con la secuencia de pasos descrita arriba. Para tener una idea general del audio correspondiente a cada video de material grabado lo que se hace es promediar el nivel de 20 paquetes de audio y eso nos da una un mejor valor representativo del nivel de audio presente en el contenido tomando en cuenta poco más de medio segundo de audio. Experimentalmente además se determinó que era mejor práctica aún almacenar hasta 10 promedios de esos grupos de 20 paquetes y al promediar estos valores se determina el nivel del audio del material grabado tomando en cuenta entonces información de hasta 200 paquetes de audio que corresponde a los últimos 6.6 segundos de audio recibidos por el servicio de interfaz con el hardware.

Para almacenar el promedio de cada grupo de 20 paquetes de audio se utiliza una cola circular de manera que eventualmente los más viejos son reemplazados por los nuevos tomando en cuenta siempre el audio más reciente recibido. Este procesamiento ha sido el que ofrece mejores resultados según las pruebas realizadas porque que el video puede contener variaciones de audio a lo largo de toda su duración pero el tomar esta cantidad de tiempo nos hace ser menos sensibles a sonidos instantáneos que generen una variación en el audio tales como truenos, golpes o efectos especiales cuyo valor repentino puede disparar el promedio de los paquetes individuales.

En el caso de la señal de estudio en cada paquete de audio recibido se realiza un proceso similar al descrito para el tratamiento del audio proveniente de materiales grabados, se tienen el mismo tipo de arreglos circulares con promedios de valores RMS de los grupos de 20 paquetes de audio y el proceso también toma en cuenta los últimos 6.6 segundos de audio en la señal de estudio para determinar el nivel de audio.

En la industria de la televisión se han ideado diferentes medidas para estandarizar el nivel de audio, debido a que en la audición humana en realidad es un parámetro subjetivo los intentos de estandarización tienen puntos a favor y en contra. Algunos estándares indican que un valor aceptable para el audio se ubica entre los -20 dBFS y -18 dBFS en promedio, esto quiere decir que si bien los picos de la señal pueden llegar incluso hasta 0 dBFS dentro de los paquetes de audio, en promedio el nivel de audio debe mantenerse en

la cifra mencionada. Estas medidas que son las más utilizadas en los estándares permiten el completo uso de la profundidad de bits sin saturar constantemente el nivel de audio.

Lo anterior tiene que ver con el concepto de rango dinámico utilizado en la ingeniería de audio así como en producción musical. Como refiere Geekland (2017), el rango dinámico es el conjunto de valores de amplitud que puede tomar una señal. Existen dispositivos en hardware y filtros digitales llamados compresores que se encargan de reducir esa cantidad de valores posibles elevando o disminuyendo en lo general el volumen perceptible para el humano. Sin embargo la percepción del audio es muy subjetiva, algunas recomendaciones indican que esto no es recomendable debido a que se pierde “viveza” en el audio ya que se reduce la cantidad de valores que puede tomar cada muestra y por lo tanto no se notan los matices del audio. Un ejemplo claro sería un sonido de un impacto súbito o un arma de fuego al ser disparada, si todo se lleva al mismo nivel no se tendría una diferencia clara entre los sonidos súbitos y fuertes y los sonidos ambientales por ejemplo.

Tomando en cuenta todo lo anterior es que se creó la recomendación en la industria de la televisión donde se indica que el nivel aceptable promedio es entre -20 dBFS y -18 dBFS, incluso está dentro de las prácticas recomendadas por estándares como ATSC. Sin embargo los picos se puede dejar que permanezcan por encima de estos valores ya que únicamente se refiere al promedio de los valores RMS.

El servicio se diseñó de manera que es posible definir un valor objetivo del nivel promedio de audio en la señal de salida en el archivo de configuración, el cual basado en la recomendación generalmente se configura a -20 dBFS, también se dio soporte para definir un valor promedio máximo aceptable que puede configurarse entre -18 dBFS y -15 dBFS permitiendo al mecanismo de normalización ser flexible y no modificar demasiado el rango dinámico del audio a la salida. Los niveles definidos en el archivo de configuración se transforman al valor real correspondiente respecto al máximo valor posible de la palabra de 16 bit al arrancar la ejecución del servicio, lo cual es posible hacer despejando de la expresión del cálculo del nivel de audio en escala dBFS el valor numérico de la muestra, los resultados se guardan para no realizar cálculos innecesarios sobre valores que son constantes debido a la naturaleza del mecanismo de normalización.

El audio de materiales grabados se normaliza de la siguiente forma:

1. Revisar el nivel del audio del archivo que fue calculado en el thread de audio en dBFS. El nivel de audio calculado en tiempo real al recibir los paquetes en el thread de audio se almacena una estructura que le permite guardar los niveles promedio de los paquetes del audio recibidos correspondientes al archivo en reproducción que se puede identificar mediante el identificador único de reproducción.
2. Comparar el nivel de audio en dBFS del video actual contra el valor en dBFS del nivel de audio objetivo, si no hay diferencia el proceso de normalización termina.
3. Despejando la expresión del cálculo del nivel en dBFS es posible llegar a una expresión que calcule a partir del nivel dBFS el valor de la muestra en términos del

máximo valor de amplitud en una palabra de 16 bits. En este paso eso es lo que se realiza, el cálculo del valor promedio del nivel de audio del archivo grabado en términos del valor máximo de la palabra de 16bits para el nivel dBFS.

4. Cálculo de un multiplicador de la amplitud, lo cual se obtiene dividiendo el valor de la amplitud del nivel objetivo entre el valor resultante del paso anterior. El multiplicador resultante puede ser mayor a 1 o menor a uno.
5. Realizar la amplificación del audio a enviar a la salida multiplicando el valor de cada muestra del paquete para todos los canales por el multiplicador obtenido. El resultado de cada operación se valida que no supere el valor máximo de la palabra de 2 bytes, de lo contrario se acota.
6. Cálculo del nivel de audio promedio para el paquete de audio resultante del paso anterior mediante el cálculo del promedio de los valores absolutos de las muestras, el valor RMS de ese promedio y el cálculo de nivel en dBFS.
7. Comparación del nivel promedio de audio del paquete de audio, llamado nivel instantáneo, contra el nivel máximo objetivo en dBFS, si resulta menor o igual al nivel máximo ya no se hace otro ajuste y el paquete está listo para enviarse a la salida con nivel normalizado. En caso contrario pasa al paso 8.
8. Dado que el nivel es más alto al permisible, se calcula de nuevo el valor real correspondiente al valor instantáneo del audio en dBFS en términos de la palabra de 2 bytes.
9. Obtener el multiplicador entre el valor instantáneo del paquete y el valor máximo objetivo.
10. Realizar la atenuación de las muestras del paquete multiplicando cada uno de los valores de las muestras por el multiplicador obtenido en el paso anterior.

El resultado de este proceso de normalización se aplica a cada paquete audio de los videos grabados que se necesita enviar salida y de esta forma mantendrán el nivel promedio de audio. Los pasos 6 al 10 sirven para validar que realmente se está cumpliendo con el objetivo de la normalización y en caso de ser necesario realizar un ajuste extra. Los procedimientos aplicados para no dejar pasar el nivel de audio más allá de cierto valor se conocen generalmente en la ingeniería de audio como limitadores o compresores dependiendo si limitan solo el valor máximo, el valor mínimo o ambos valores del nivel de audio de la señal. En el desarrollo de este mecanismo de normalización se tomaron en cuenta varias ideas del ámbito de la producción musical.

El procedimiento para determinar el nivel del audio en la señal externa es similar al descrito para el audio de materiales grabados, la diferencia es que el nivel del audio de la señal externa se calcula directamente en el callback del dispositivo de entrada para cada paquete y se determina también el promedio del nivel de la señal. Al igual que los videos grabados se cuenta con estructuras de datos que permiten almacenar y acceder a esta información, el mecanismo como ya se expuso anteriormente toma en cuenta alrededor de 6 segundos de audio proveniente del estudio y ocupa el valor RMS del nivel promedio de la señal.

La normalización de nivel de audio, tanto para la señal externa como del material grabado, se realiza en el objeto manejador de datos de salida con los pasos descritos anteriormente en tiempo real. Los canales destinados a la inserción de cuetones son descartados de este tipo de proceso por eso se introducen después de la normalización.

El procesamiento de audio y el de video se realiza de manera simultánea en el manejador de datos de salida, todas las operaciones necesarias se deben realizar en máximo 33 milisegundos para cada cuadro de la señal incluyendo el envío de los datos de audio y video al buffer del hardware.

Control de mecanismo de protección bypass

Los sistemas de automatización operan las 24 horas todos los días del año por lo que deben contar con algún tipo de mecanismo de redundancia en caso de presentarse una falla en alguno de sus componentes. En este tipo de sistemas puede haber fallos en el software o en los componentes del hardware tales como tarjetas de video, tarjetas madre, procesadores, módulos de memoria, discos duros, hardware para la captura y reproducción de la señal o una fuentes de alimentación. Aunque se diseñó un plan de mantenimiento preventivo puede presentarse eventualmente un problema y es necesario que ante las emergencias no se interrumpa la señal.

El modelo de tarjeta utilizado BlackMagic Decklink Quad contaba con un mecanismo de protección basado en un “watchdog”, el cual en cuanto detecta cierta condición conecta la entrada de una tarjeta particular con su correspondiente salida mediante un relevador logrando así enviar la señal de la tarjeta de entrada a la tarjeta de salida. El conectar la señal de entrada a la salida del mismo dispositivo se le conoce como estar en bypass. Muchos dispositivos en televisión cuentan con la característica de bypass, si tienen entradas y salidas, en muchos de ellos la señal de la entrada es transmitida automáticamente a la salida del dispositivo.

Cada tarjeta BlackMagic Decklink Quad tenía 4 tarjetas independientes y cada una de éstas a su vez tenía una entrada y una salida para señal SDI, de forma que cada una de las 4 tarjetas puede ser conectada en modo bypass.

El fabricante indicaba que el desarrollador debía activar el mecanismo de watchdog en algún momento de la ejecución de la aplicación. Para activar este mecanismo el SDK cuenta con una función que recibe un número entero que representa el tiempo máximo de espera en milisegundos del watchdog. Esta función debe ser llamada enviando como argumento el mismo valor de tiempo u otro diferente antes de que pase el tiempo que se había indicado en milisegundos en la primera llamada, de lo contrario si transcurre la cantidad de tiempo programado sin que se vuelva a reportar la aplicación entonces se activa el relevador de las tarjetas y entran a bypass. Para que las tarjetas no entren en estado de bypass la aplicación debe reportarse entonces continuamente enviando un nuevo valor de tiempo de espera al watchdog.

En términos prácticos el tiempo indicado en la llamada de la función que activa el watchdog es el tiempo máximo para que la tarjeta se conecte en bypass. Esta característica se puede habilitar para todas las tarjetas o bien solamente para algunas.

En el servicio de interfaz con el hardware el tiempo máximo de espera del watchdog se puede definir en el archivo de configuración, así como también si se desea o no activar esta característica. Si el soporte para el watchdog está activo entonces al iniciar la reproducción en el hardware el servicio envía al hardware el tiempo de espera del watchdog definido en el archivo de configuración, el cual por default es de 500 ms. En caso que los dispositivos de salida se encuentren en bypass el enviar un tiempo de watchdog al hardware e iniciar la reproducción elimina el estado de bypass.

Cuando la reproducción se está realizando en cada callback del hardware de salida, ya que se enviaron tanto audio como video a los dispositivos de salida, se actualiza el tiempo del watchdog en los dispositivos de salida y siguen funcionando sin entrar en estado de bypass. El watchdog es un mecanismo que permite que las tarjetas de salida entren en estado de bypass ante cualquier falla en el software e incluso en el hardware ya que está diseñado por el fabricante de manera que se active aunque haya pérdida de energía o el equipo se haya apagado.

Para aprovechar esta característica se conectan dos equipos de cómputo con el módulo de reproducción en cascada. Las tarjetas de salida del equipo de respaldo se conectan a las entradas del equipo principal correspondientes a los dispositivos de salida de éste último. De esta forma cuando el equipo principal falla después de 500ms las tarjetas de salida entrarán en bypass y estará automáticamente al aire el equipo de respaldo.

El equipo de respaldo puede o no tener esta característica activada, si no la tiene activada en caso de presentarse una falla en ambos equipos la señal de salida del sistema contendría negros, pero si el soporte del watchdog está activo también en el equipo de respaldo se le puede conectar otra señal de respaldo proveniente de algún otro dispositivo.

La conexión en cascada no impide que el equipo principal pueda ser desconectado del tren de transmisión ya que se instalaron paneles de parcheo que permitían evitar el paso de la señal de salida por el equipo principal permitiendo conectar directamente la salida del equipo respaldo a los siguientes dispositivos de distribución, también dependiendo de la falla pudiera no ser necesario desconectarlo del tren de transmisión o bien en otros casos sería posible esperar a un horario de mantenimiento adecuado para realizar la reparación o reemplazar el equipo.

Al conectar en cascada los equipos en el caso que cuenten con salidas en formato HD y SD simultáneas se debe cuidar la correspondencia de los formatos en la conexión de salidas y entradas del equipo de respaldo al equipo principal.

Estas son todas las características del servicio de interfaz con el hardware, realicé el desarrollo del mismo en un tiempo de 2 meses, como se expuso se han agregado nuevas

características y optimizaciones principalmente en el mecanismo de combinación de sùpers y diferentes opciones de manejo de logos.

Las tareas que desempeñé abarcan el diseño del servicio, la investigación de las técnicas descritas, programación del código, pruebas e implementación. El servicio de interfaz con el hardware fue escrito en lenguaje de programación C++ con la integración de las diferentes funciones de la librería libav para la manipulación de escalamientos, transformación de espacios de color de pixeles y la integración del SDK del hardware utilizado.

El diseño basado en componentes para el módulo de reproducción permite la posibilidad de reemplazar el Reproductor de Máster por otra versión u otra aplicación sin la necesidad de modificar la capa de software que se comunica directamente con el hardware. También podría mejorarse esta capa de software de comunicación con el hardware sin realizar cambios al Reproductor de Máster que tiene la lógica de ejecución de la guía de programación. Como trabajo futuro y áreas de oportunidad se ha planeado dar soporte para más transformaciones gráficas, efectos de transición, la combinación de más elementos gráficos en pantalla además del logo y los sùpers, esto es llamado en los sistemas de automatización canales gráficos. También contar con un mecanismo para recargar del logo sin necesidad de reiniciar el servicio. En el caso de reinicio del servicio se cuenta con el mecanismo de respaldo automático por lo que la señal de salida no se ve interrumpida en ningún momento.

3.8.2 Reproductor de Máster

El servicio de interfaz con el hardware descrito en la sección anterior se encarga de enviar la información de audio y video a los dispositivos de salida además de gestionarlos, pero por otro lado fue desarrollado un componente del sistema que lleva el control del orden y el tiempo de ejecución de las operaciones así como la secuencia de eventos que componen la guía de programación esta aplicación se llama Reproductor de Máster.

El Reproductor de Máster es una aplicación que desarrollé completamente en lenguaje C++ y la integración de otras librerías, esta aplicación es la interfaz de usuario del módulo de reproducción del sistema de automatización. El Reproductor de Máster se encarga de controlar el servicio de interfaz con el hardware mediante el envío de los diferentes comandos soportados, el envío de la información de audio y video decodificada de los materiales grabados así como el manejo de la guía de programación.

Los requerimientos más importantes solicitados para el Reproductor de Máster fueron los siguientes:

- Funcionamiento ininterrumpido las 24 horas del día todos los días del año
- Los canales con contenido 100% grabado deberían contar con un reproductor totalmente automatizado:

- Las únicas intervenciones necesarias por parte de un operador deberían ser necesarias para mandar a corte comercial los programas en vivo en los canales que manejen este tipo de contenido o bien para realizar modificaciones en los eventos de la guía.
- Permitir pre visualización de materiales grabados
- Posibilidad de programar súpers en tiempos de reproducción definidos previamente en la guía de programación sobre bloques de programa.
- Descarga diaria automática de la guía de programación.
- Revisión automática de materiales a transmitir.
- Generación de reportes de transmisión

Las principales funciones del Reproductor de Máster que se definieron son las siguientes:

- Interfaz gráfica del módulo de reproducción para el operador de máster
- Carga, administración y ejecución de la guía de programación
- Descarga de los videos de los servidores de almacenamiento
- Envío de comandos de control al servicio de interfaz con el hardware
- Recepción del estado del servicio de interfaz con el hardware y su representación
- Decodificación de los archivos de audio y video
- Decodificación de los súpers
- Pre visualización de materiales grabados a transmitir
- Revisiones automáticas de material a transmitir
- Envío de alertas automáticas de diagnóstico
- Sincronización de operaciones con reproductor máster de respaldo

La arquitectura del reproductor de máster tiene como base una interfaz principal, que es además el principal hilo de ejecución de la aplicación, desde la cual el operador puede acceder a otras ventanas para realizar acciones relacionadas con la administración de la guía de programación, la visualización del material en forma de previo y otras más. El reproductor de máster cuenta además con un objeto que gestiona la conexión a la base de datos, un objeto para el envío de alertas mediante correo electrónico, diferentes threads de ejecución independientes de la interfaz gráfica principal para realizar sus tareas en forma simultánea, un objeto para almacenar la guía de programación, buffers para almacenar la información decodificada de los diferentes materiales grabados y pipes de datos que permiten enviar tanto los comandos como los datos decodificados de audio y video al servicio de interfaz con el hardware.

La arquitectura completa se muestra en el siguiente diagrama:

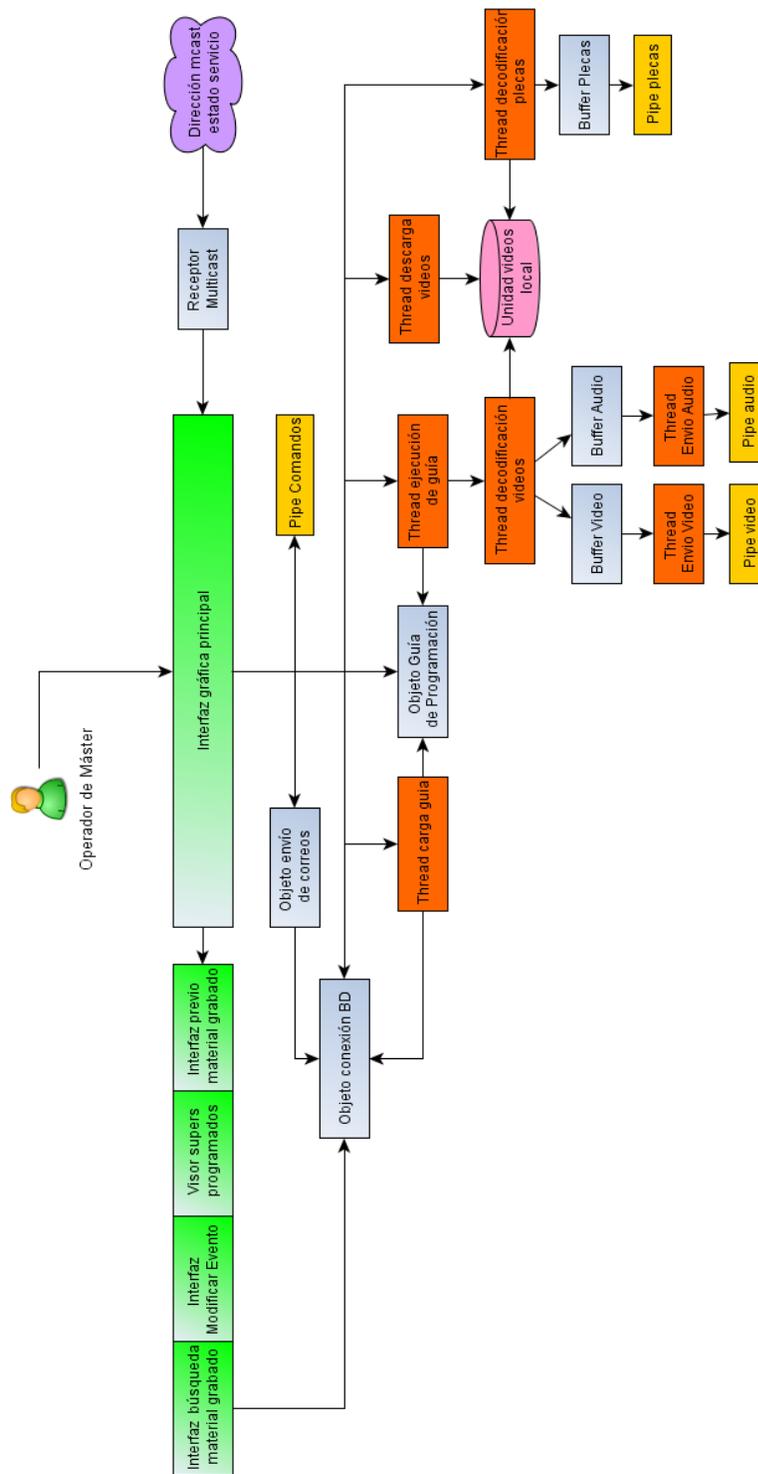


Figura 14 Arquitectura del reproductor de máster

Interfaz principal del Reproductor de Máster

La interfaz principal del Reproductor de Máster incluyó los siguientes elementos:

- Un menú principal de aplicación que cuenta con las opciones Guías y Cintillos.
- Barra de estado del decodificador donde muestra la actividad actual del reproductor, un reloj de cuenta regresiva para el evento actual grabado, duración restante de la guía cargada y si está en bloque de comerciales (carrier) un reloj con cuenta regresiva con el tiempo restante del bloque.
- Botón para reconectar la Base de Datos
- Panel con indicadores del estado de la base de datos y estado de la conexión a un reproductor de respaldo
- Barra con la información del evento actual al aire
- Sección de controles para modificar la guía de programación
- Barra de estado de descarga de videos
- Una tabla con los eventos de la guía de programación
- Controles de reproducción para iniciar, detener y avanzar al siguiente evento de la guía de programación
- Indicadores de actividad del servicio de interfaz con el hardware y la conexión del reproductor con el servicio.
- Controles manuales para activar o desactivar la inserción del logo en el servicio de interfaz con el hardware.
- Controles manuales para cargar y reproducir sùpers

La aplicación se instaló en los equipos disponibles con sistema operativo Windows 7 y podía contar o no con un usuario operador de máster, esto depende del tipo de eventos que pueda incluir la guía del canal. Se soportaron entonces dos tipos de canales:

- Canal automatizado. En este tipo de canales la transmisión es automática y todos sus contenidos son grabados, la guía que se entregaba un día anterior a su transmisión y rara vez requería cambios, si es necesario realizarlos se deberían hacer con la intervención de un operador.
- Canal semi-automatizado. Es un canal donde se transmitirían eventos tanto grabados como provenientes de una señal externa, es decir en vivo. Requería un operador de máster que interactuara con la interfaz gráfica del reproductor para realizar los ajustes en la guía derivados del tiempo variable de transmisión de cada bloque de programa en vivo y para mandar a corte comercial la señal. El operador tenía comunicación directa con el estudio en vivo mediante un intercom donde recibiría los conteos para mandar a corte comercial.

Se definió que los operadores utilizarían esta interfaz principal para realizar las tareas de administración de la guía, enviar la señal a corte comercial y conocer el estado del módulo de reproducción. En caso de un fallo fatal del software las aplicaciones se diseñaron de tal forma que en caso de requerir el reinicio del servicio de interfaz con el hardware así como el reinicio del Reproductor de Máster, es posible recargar la guía de programación e

iniciar su reproducción desde el evento donde se hubiera interrumpido con el objetivo de restablecer la operación.

La interfaz principal permitía acceder a las otras interfaces o ventanas secundarias expuestas en el diagrama de bloques de la aplicación tales como el visor de previo, el visor de súpers programados, la interfaz para modificar eventos y la interfaz para la búsqueda de materiales grabados en los servidores de almacenamiento. Las ventanas secundarias se mostraban superpuestas sobre la ventana principal pero detenían las actividades del hilo principal de ejecución y mucho menos de los hilos secundarios.

Como se muestra en el diagrama, la interfaz principal utiliza los demás componentes para realizar las diferentes tareas que debe ejecutar la aplicación, algunos son objetos, threads, estructuras de datos o buffers. La interfaz principal además recibía mediante un objeto de manejo de tráfico multicast el estado del servicio de interfaz con el hardware, lo que le permite representar el estado del servicio a la vista del usuario así como mostrar el tiempo restante de reproducción del evento grabado actual. Los eventos en vivo no tienen una duración estimada exacta debido a que podían tener cualquier duración que el productor en el estudio va determinando, por lo que no era factible en esos casos colocar el reloj de cuenta regresiva.

La sincronización de los hilos de ejecución del control avance de la guía, carga de la guía y descarga de videos se integró en el hilo principal de ejecución que incluye la interfaz gráfica principal. Las tareas de decodificación se realizan en subprocesos debido a la capacidad de procesamiento requerida y que esta tarea no debe ser interrumpida. El hilo principal de ejecución también contiene el extremo del pipe de comandos desde donde se envían los buffers de bytes con los datos de los comandos expuestos en el servicio de interfaz con el hardware.

A continuación se exponen los principales elementos de la interfaz gráfica principal así como las funciones que el usuario puede ejecutar a través de estos controles, después de esta sección se detallarán las actividades automáticas que se realizan en la interfaz gráfica y posteriormente se abordará la interacción con los demás elementos que componen la aplicación.

Menú principal de la aplicación

Cuenta con dos opciones Guías y Cintillos.

La opción de guías contiene a su vez las siguientes opciones:

- Cargar guía de programación: Permite borrar la guía actual totalmente para cargar una nueva guía de programación contenida en un archivo que el usuario selecciona a través de un diálogo. Si la aplicación se encuentra en reproducción, entonces la reproducción se detiene para borrar la guía actual de memoria.

- Añadir guía de programación: Añade al final de la guía actual una guía de programación contenida en un archivo que el usuario selecciona. Si la aplicación está en reproducción ésta no se detiene.
- Cargar guía temporal: El Reproductor de Máster se diseñó para respaldar periódicamente la guía de programación que está ejecutando con todos los cambios realizados y poder recuperarla en caso de que sea necesario cerrar el reproductor. Esta opción permite al operador recuperar la guía temporal que necesite, la extensión de estos archivos es .tmp. La guía temporal puede ser copiada manualmente desde el Reproductor de Máster de respaldo en caso de que la falla dure horas para restablecer el equipo principal con todos los cambios.
- Descargar guía del día: Las guías se generaban en el sistema de tráfico y eran depositadas en repositorios dentro de un servidor. Esta opción permite al reproductor conectarse mediante FTP a ese servidor, los datos de la cuenta de FTP están definidos en el archivo de configuración así como la dirección IP y puerto de conexión del servidor FTP. El nombre de los archivos de las guías incluyó la fecha y un identificador del canal para llevar el control. El reproductor de máster además puede ser configurado para que una vez descargada la guía sea añadida automáticamente a la guía actual sin interrumpir la reproducción o solamente la descargue al disco duro del equipo.
- Descargar guía siguiente día: Similar a la opción anterior pero descarga la guía de programación del día siguiente a la fecha actual.
- Enviar ASRUN: El Reproductor de Máster genera un reporte de los materiales reproducidos, estos reportes se guardan en archivos llamados asrun. Los asrun son archivos de texto que contienen información para el sistema de tráfico. Los reportes contienen el resultado de la reproducción, la hora exacta de inicio y otros datos que permitían al sistema de tráfico identificar si los eventos de la guía de programación planeada fueron transmitidos al aire exitosamente o no. Esta opción permite al operador enviar mediante FTP los archivos asrun al servidor de donde descarga las guías planeadas. También es posible configurar el reproductor de máster para que automáticamente envíe estos reportes al servidor designado cuando realiza el cambio de guía.

El menú de cintillos contaba con una única opción Ver Programados, lo que hace es abrir la interfaz del visor de sùpers programados, esta es una ventana con una tabla que muestra los bloques de programa en los cuales están programados cintillos y en qué tiempos relativos al inicio de cada bloque está programada su reproducción.

Barra estado del decodificador

La barra principal de estado tenía 4 elementos: la etiqueta de estado, reloj de cuenta regresiva del evento actual grabado, duración acumulada de la guía y la cuenta regresiva del carrier.

Etiqueta de estado. Informa al usuario acerca del estado del decodificador de video, puede tener alguna de las siguientes leyendas:

- Esperando: El reproductor acaba de arrancar y aún no ha realizado acción alguna.
- Reproduciendo: El Reproductor de Máster se encuentra decodificando un material grabado y enviando audio y video al servicio de interfaz con el hardware.
- Cambio: El usuario ha solicitado pasar al siguiente evento mediante los controles de reproducción.
- Fuente externa: El decodificador no está activo, se está enviando en el servicio de interfaz con el hardware video proveniente de la señal externa a la salida.
- Detenido: Se ha detenido el proceso de reproducción por una falla grave o a petición del operador.
- Cuenta regresiva: Leyenda “Cambio” acompañada del número de segundos restante para que un nuevo evento se emita al aire. Cuando el usuario inicia reproducción o bien solicita cambiar al siguiente evento se tiene una cuenta regresiva de 3 segundos para que el cambio se vea reflejado la señal de salida. Lo anterior es acostumbrado en televisión para prevenir a los conductores o sincronizar acciones en los eventos en vivo tanto en el estudio como en las cabinas de máster.

La leyenda a mostrar en la etiqueta de estado se define de acuerdo a la información recibida del servicio de interfaz con el hardware a través del tráfico multicast que emite el servicio. La recepción y tratamiento de esta información se lleva a cabo mediante el objeto receptor multicast.

Reloj cuenta regresiva evento actual. Muestra un reloj en cuenta regresiva que se activa al iniciar un evento grabado. El valor inicial del reloj corresponde a la duración total detectada del material y va disminuyendo a cero conforme el servicio de interfaz con el hardware reporta que ha transmitido a la salida los cuadros de video del material. Por eso la estructura de datos de cada cuadro enviada mediante el pipe de videos lleva el identificador de reproducción del archivo al que corresponde. El reloj era útil para que el operador de máster conozca el tiempo restante del evento actual.

Reloj duración acumulada. Muestra el total de horas, minutos, segundos y cuadros de los eventos de la guía que aún están por ser transmitidos. Sirve para verificar el tiempo total restante de la guía de programación, al cargar la guía también era útil para saber si la guía se cargó completa.

Reloj cuenta regresiva carrier. Los carriers son el conjunto de promos y comerciales entre dos bloques de programa. Este reloj muestra al operador el tiempo que falta para llegar al siguiente bloque de programa sumando la duración todos los eventos del carrier que quedan por transmitir y el tiempo restante del evento actual si es que es parte del carrier. Este reloj solamente se activa cuando la aplicación reproduce un evento que no es bloque de programa, los cuales están identificados en la interfaz gráfica.

Botón Conexión BD

El botón de conexión BD sirve para que la aplicación ejecute una re conexión al servidor de base de datos del sistema. Si el reproductor de máster ya está conectado al servidor

de base de datos cierra esa conexión y abre una nueva. Las conexiones con el servidor se gestionan a través del objeto de conexión con la base de datos.

Panel con indicadores del estado de la BD y conexión a reproductor de respaldo

Este panel contiene 3 indicadores útiles para el operador:

- Indicador de conexión con base de datos principal. El Reproductor de Máster se diseñó con capacidad para utilizar dos servidores de base de datos. Las consultas se realizan siempre al servidor principal y en caso de fallo se recurre al de respaldo. Este indicador adquiere el color verde si se establece la conexión y las consultas son exitosas.
- Indicador de conexión con base de datos respaldo. Es posible configurar un servidor de base de datos de respaldo. Se mantiene en color rojo si no se establece la conexión o cuando el soporte para el servidor de base de datos de respaldo no está activo en el archivo de configuración.
- Indicador Estado Conexión: Informa al usuario el estado de la conexión con un Reproductor de Máster de respaldo. La sincronización incluye las operaciones para borrar, modificar y agregar eventos a la guía de programación así como el paso de bloques de programa en vivo a cortes comerciales en los canales que lo requieren, de esta forma el operador solamente debe manipular uno de los dos equipos reproductores. El indicador es verde si la conexión está activa, gris si no fue configurada y rojo si hay un problema.

Barra con la información del evento actual al aire

La guía de programación diariamente incluye generalmente mínimo 100 eventos hasta un máximo de 1500 dependiendo del canal y día de la de la programación. La guía de programación se muestra en un elemento que se expondrá un poco más adelante en el informe en forma de una tabla. La barra de evento actual fue diseñada para mostrar siempre a la vista del usuario el evento actual en reproducción con los mismos datos incluidos en la guía de programación para efectos de una visualización rápida.

Controles para modificar la guía de programación

La interfaz gráfica incluye botones acomodados en una barra debajo de la información del evento actual que permiten al operador realizar modificaciones a la guía de programación. Cada botón además está asignado a una tecla de función para su rápida operación. Los botones disponibles son:

- Agregar video. Permite agregar un evento con fuente tipo disco (material grabado) antes de la posición actual seleccionada sobre la guía recorriendo el evento seleccionado y los subsecuentes hacia abajo. Para seleccionar el material se abre la interfaz de búsqueda de material grabado.
- Eliminar evento. Permite eliminar un evento de la guía de programación

- Reproducir aquí. Si el reproductor está detenido y además tienen una guía cargada permite al operador iniciar la reproducción desde el evento seleccionado en la guía. Al presionarlo la aplicación muestra un mensaje de confirmación el cual puede ser utilizado para arrancar la reproducción en el momento preciso.
- Agregar video siguiente: Inserta un evento con fuente tipo disco, es decir material grabado, después de la posición actual seleccionada sobre la guía recorriendo los subsecuentes hacia abajo.
- Verifica video. Revisa que el código de material del evento seleccionado en la guía sea el correspondiente al código de inventario mostrado. Para determinarlo realiza una búsqueda en la base de datos tomando como referencia el código de inventario, si obtiene un código de material válido esto indica que el material ya fue ingestado a los servidores de almacenamiento y entonces revisa que el material ya se encuentre en el disco local, en caso de no contar con el material solicita su descarga al thread de descargas. Al finalizar la operación actualiza la visualización del registro seleccionado ya sea a un estado de error, de descarga pendiente o bien si ya estaba en el disco duro local cambia al estado cargado.
- Agregar video final. Agrega un evento con fuente tipo disco al final de la guía.
- Actual. Posiciona el renglón actual de la tabla en el registro que corresponde al evento actual al aire.

Barra de estado de descarga de videos

La barra de estado de descarga de video muestra el progreso de descarga de un material desde el servidor de almacenamiento indicando el nombre del archivo y el porcentaje de avance. En el caso de los videos que se ingresarían al sistema para eventos muy próximos de transmisión al aire era indispensable que el operador conociera el estado de avance de la descarga. Las descargas son atendidas por el thread de descargas que reporta sus acciones y avance mediante mensajes del sistema operativo a la interfaz gráfica.

Guía de programación

La guía de programación es la secuencia de eventos que se transmiten a lo largo del día en el sistema de automatización. La secuencia se conforma de bloques de programa, cuya fuente puede ser una señal externa o un programa grabado en un archivo, entre estos bloques de programa generalmente se encuentran uno o más eventos que corresponden al corte comercial y son de fuente grabada, es decir, son archivos en el disco duro del equipo.

En la interfaz gráfica del reproductor de máster la guía de programación se muestra en una tabla donde cada renglón representa un evento de la guía y las columnas tienen información relevante de cada evento. Los datos desplegados en forma de columnas de la tabla para cada evento son:

- Número de renglón
- Hora de inicio

- Nombre del evento
- Código de inventario
- Código de Material
- Duración del archivo si la fuente es grabada
- Duración planeada en la guía de programación
- Duración real al aire una vez que ha sido reproducido
- Tipo de Evento, indica si es un comercial o bloque de programa
- Fuente del evento, indica si es de disco duro o fuente externa
- Logo, indica al usuario si la inserción de logo se activa automáticamente en ese evento
- Identificador del evento: Muestra un identificador único para efectos de control. En la sección de carga de guía se detalla su uso y generación.
- Estado. Indica el estado del evento por ejemplo si está cargado y listo para salir al aire, si el video está siendo descargado, el video no existe en el servidor de almacenamiento, si ya pasó al aire, es el actual, está en estado de pre-roll o tuvo algún problema para ser transmitido.
- Cuetone Inicio. Indica si en ese evento está programado insertar en la señal un cuetone de inicio de bloque de comerciales donde se autoriza bloqueo. El cuetone de inicio se debe reproducir 3 segundos antes de cambiar al siguiente evento.
- Cuetone Fin. Indica si en el evento está programada la inserción de un cuetone de fin de un bloque de comerciales donde se autorizó el bloqueo. También se reproduce 3 segundos antes de finalizar el evento en el cual está programado.

Con estos datos el usuario tenía la información necesaria para revisar la correcta ejecución de la guía. La tabla en la interfaz se actualiza cuando hay algún cambio en el estado de los eventos por ejemplo cuando cambia el evento actual, el estado del video grabado se ha modificado a listo, o se está descargando, también puede ser actualizada cuando el usuario realiza modificaciones a la guía con los controles correspondientes para que tenga una retroalimentación de los cambios que ha efectuado. La tabla que contiene la guía también se actualiza cuando se carga una nueva guía o bien al añadir una guía, en este caso la actualización es paulatina y se va mostrando en la interfaz gráfica grupos de eventos que van siendo agregados a la guía lo cual impide que se bloquee el reproductor de máster y además permite que el operador pueda seguir utilizando los demás componentes de la interfaz en caso de ser necesario.

Controles de reproducción

Los controles de reproducción se ubican debajo de la guía de programación en la interfaz gráfica principal, son 3 botones que permiten al operador realizar las siguientes acciones:

- Inicio de reproducción. Permite al usuario iniciar la reproducción de la guía desde el primer evento.
- Detener reproducción. Al activarlo se detiene la lista de reproducción así como el envío de información al hardware de salida en el servicio de interfaz con el hardware.

- Siguiendo evento. Realiza el cambio del evento actual al siguiente en la guía, esta operación incluye una espera de 3 segundos con conteo regresivo en la barra de estado en la cual la aplicación prepara el siguiente evento y además el operador puede utilizarlo para estar en sincronía con otras actividades.

Los controles de reproducción se encuentran deshabilitados por default en la interfaz gráfica y se habilitan solamente si el operador mantiene presionada la tecla de CTRL. Para utilizarlos es necesario presionar dicha tecla y mientras está presionada dar clic en el botón, esta es una medida de protección para la operación ya que las acciones ejecutadas a través de estos controles afectan la reproducción de la guía.

Indicadores de actividad de servicio de interfaz con el hardware

En la esquina inferior izquierda de la interfaz gráfica principal se muestran una serie de indicadores, los cuales muestran el estado del servicio de interfaz con el hardware además del estado de la decodificación de videos realizada por el reproductor de máster.

El reproductor de máster cuenta con los siguientes indicadores:

- Estado de la conexión con los pipes del servicio de interfaz con el hardware. Indican si está activa la conexión con los pipes de Video, pipe de Audio y pipe de comandos que son necesarios para llevar a cabo la comunicación con el servicio de interfaz con el hardware. Gráficamente son un conjunto de etiquetas que parpadean en color verde si todo se encuentra funcionando bien, en caso contrario parpadean en color rojo.
- Información del video actual en decodificación que el servicio de interfaz con el hardware está enviando a la salida. Son un conjunto de 4 cajas de texto que muestran los datos relevantes del video que se está enviando al servicio de interfaz con el hardware y a su vez son enviados a la salida. Los datos mostrados en las cajas de texto incluyen un identificador único que se genera en tiempo de reproducción para control, el número de cuadro que se está enviando al hardware de salida, el identificador del archivo anterior reproducido y la hora en milisegundos en la cual inició el envío de datos al hardware de salida.
- Estado del servicio. La interfaz gráfica principal cuenta con 3 cajas de texto que contienen un resumen del estado del servicio. Una de estas cajas de texto contiene el código numérico correspondiente al estado del servicio que indica si está enviando datos a la salida de un video grabado, si la señal externa se emite a la salida, si se encuentra esperando instrucciones o tiene algún error. La segunda caja muestra el código de estado de la reproducción de un video de disco y puede indicar si está en reproducción, en proceso de cambio de video o un error. La tercera caja muestra el nivel del buffer del hardware de salida para detectar alguna anomalía en el funcionamiento global del módulo de reproducción.
- Contadores de paquetes decodificados de audio y video. Un conjunto de cajas de texto que indican cuántos paquetes de audio y de video se han logrado decodificar del archivo actual de disco si es que el evento es de fuente grabada. También

muestra cuántos paquetes están almacenados en los buffers de la aplicación reproductor de máster a la espera de ser enviados al servicio de interfaz con el hardware y un resumen de los cuadros que se han decodificado con errores si es que ha sucedido.

La información del estado es enviada desde el servicio de interfaz con el hardware mediante un mecanismo de tráfico de red multicast, el Reproductor de Máster cuenta con un componente receptor de tráfico multicast. Cada paquete recibido a través de este mecanismo incluye la información cuadro a cuadro de los datos que está enviando el servicio de interfaz con el hardware a la salida en el caso de estar reproduciendo un archivo almacenado en el disco duro del equipo, por el contrario si el servicio se encuentra enviando señal externa al hardware de salida entonces la notificación vía multicast es generada cada medio segundo.

La información referente a la decodificación es generada en el mismo Reproductor de Máster mediante mensajes del sistema operativo provenientes de los threads dedicados a la decodificación de audio y video.

Los indicadores de esta sección de la interfaz gráfica se diseñaron para diagnóstico y revisiones por parte del personal de soporte en caso de que se presente un mal funcionamiento del sistema. El Reproductor de Máster mediante la recepción de información proveniente del servicio de interfaz con el hardware y su interpretación es capaz de conocer el estado de la reproducción en los dispositivos de salida, como se expondrá más adelante la interpretación de esta información permite la sincronización de algunas acciones y se utiliza para mostrar los relojes de cuenta regresiva en la interfaz gráfica principal.

Controles manuales de logo

Dentro de la interfaz gráfica principal debajo de la guía de programación, en la parte central se encuentran dos botones que le permiten al operador activar o desactivar inmediatamente la inserción del logo en la señal de salida enviando los comandos correspondientes al servicio de interfaz con el hardware.

Controles para reproducción de súper

En la parte inferior derecha de la interfaz gráfica principal se tienen los controles e indicadores suficientes para ejecutar de forma manual un súper o cintillo. Los elementos de la interfaz gráfica para este fin son:

- Indicador de pipe de plecas. Es una etiqueta que indica el estado de la conexión con el pipe de plecas cuyo extremo servidor está en el servicio de interfaz con el hardware, parpadea en verde si la conexión es correcta y en rojo si hay un problema con el pipe.
- Botón Seleccionar pleca. Al dar clic sobre el botón se despliega una ventana donde el operador puede buscar la pleca por código de inventario o código de

material, la interfaz gráfica es similar a la que se utiliza para buscar un video grabado.

- Botón Reproducir pleca. Una vez que el operador ha seleccionado la pleca a reproducir, este botón le permite iniciar la reproducción de la pleca, lo que incluye la decodificación y el envío de los cuadros decodificados al servicio de interfaz con el hardware para su inserción en la señal de salida.
- Código de inventario. Las plecas como todos los materiales almacenados en el sistema cuentan con un código de inventario, una vez que el usuario selecciona manualmente uno se muestra en una caja de texto el código de inventario seleccionado. De esta manera si el operador lo desea puede verificar que el material sea el adecuado antes de reproducirlo. Si hay plecas o cintillos programados en esta caja de texto aparece el código de inventario del siguiente material a reproducir.

Con este último grupo de elementos de la interfaz principal termina la mención de todos los elementos que la conforman.

Interfaces secundarias del reproductor de máster

El reproductor de máster cuenta además con interfaces gráficas secundarias a las cuales se accede mediante algunos controles para realizar tareas específicas sobre la guía de programación.

Interfaz previo de material grabado

La interfaz previo de material grabado es una ventana secundaria a la cual se accede dando doble clic sobre un registro de la guía de programación en la interfaz gráfica principal cuya fuente sea "DISCO", lo cual indica que es un material grabado que el reproductor ha descargado previamente del servidor de almacenamiento.

Si la fuente no está marcada como disco sino EXTERNA o el estado indica que el video no se encuentra en el disco duro del equipo entonces la aplicación emite un beep.

Al dar doble clic sobre el registro se abre esta ventana secundaria que es un reproductor desarrollado con la integración de las librerías de libav para cargar el video y visualizarlo en la ventana. Cuenta además con los controles básicos de reproducción para iniciar la reproducción, hacer una pausa, adelantar o atrasar un cuadro en el video y también una barra de desplazamiento que permite avanzar a cierta posición del video. En la esquina inferior derecha tiene un contador que indica el avance de la reproducción en timecode así como la duración total del video cargado en este reproductor.

Esta ventana se diseñó para que el operador pudiera revisar de ser necesario si el material que se indica en la guía es el correcto además de verificar algunas otras cosas como la calidad del audio o del video antes de ser transmitido al aire. También podía llegar a haber casos donde el material enviado desde las cabinas de edición no fuera el correcto o se tenía la duda de que el material enviado al sistema fuera el adecuado, en

estos casos también es indispensable poder monitorear el material antes de la transmisión.

La interfaz de previo es una ventana flotante sobre la ventana principal y su visualización no detiene la ejecución de la guía ni afecta la reproducción de los videos aún cuando el video cargado en este reproductor sea el siguiente o bien inicie su reproducción dentro del Reproductor de Máster. La única restricción que tiene el Reproductor de Máster es no permitir la reproducción de eventos pasados, debido a que para esa operación existen otros visores de materiales en otros componentes del sistema, por otro lado ya que el evento es pasado no vale la pena invertir los recursos del Reproductor de Máster cuando el problema no es posible solucionarlo ya desde esta aplicación.

Visor de súpers programados

El visor de súpers programados es una ventana secundaria del Reproductor de Máster a la cual se puede acceder mediante el menú de cintillos en la opción ver cintillos o súpers programados. Como se ha mencionado en este reporte las palabras súper, pleca y cintillo son sinónimos dentro del mundo de la televisión y todos ellos son materiales sobrepuestos en la señal de salida que genera módulo de reproducción.

Uno de los requerimientos del Reproductor de Máster fue el poder programar la reproducción de súpers sobre bloques de programa desde el momento de la creación de la guía de programación para evitar operaciones manuales desde luego sujetas al error humano y para las cuales se requiere un operador. Este requerimiento fue planteado meses después de llevar el sistema a producción en una tercera etapa de desarrollo.

Para dar soporte a este requerimiento se trabajó conjuntamente con el área de desarrollo que tiene a su cargo el sistema de tráfico donde se generan las guías de programación. La guía de programación era un archivo de texto plano con los datos que componen la guía y se diseñaron para contener los campos necesarios en los que se indica que un súper debe ser transmitido sobre cierto bloque de programa así como el tiempo transcurrido desde el inicio del bloque en el cual debe ser reproducido el súper. Como restricción solamente se puede transmitir un súper a la vez. Este requerimiento se desarrolló para mostrar en pantalla por ejemplo un letrero que indique el nombre del programa actual, la clasificación de audiencia del programa o alguna otra información importante.

El visor de cintillos programados es una ventana con una tabla que muestra los súpers programados en el sistema con los siguientes datos:

- Nombre del material
- Código de inventario
- Código de material
- Número de evento del bloque de programa sobre el cual se va a reproducir
- Tiempo transcurrido desde el inicio del bloque para el inicio de la reproducción del súper

- Duración del súper
- Estado del material, si se tiene en disco duro, se va a descargar, está listo para ser reproducido o ya fue transmitido el aire.

La información de la pauta de los cintillos programados está almacenada internamente en una estructura de datos tipo lista que mantiene todos los datos visualizados en la tabla y algunos otros utilizados para el control de la reproducción de los súper. Esta lista contiene estructuras para representar cada uno de los cintillos programados, el visor expuesto en este punto solamente se diseñó para mostrar al usuario la información con la finalidad de conocer los materiales programados en la guía y si se han reproducido adecuadamente durante el día.

Interfaz modificar evento

La interfaz modificar evento es otra ventana secundaria a la cual el operador puede acceder desde el botón modificar evento o bien al seleccionar la opción modificar evento desde el menú contextual que tiene la guía de programación en la interfaz principal.

Al abrir la ventana despliega los datos actuales del evento seleccionado para modificarlo entre los cuales se encuentran la hora de inicio, nombre, duración, el tipo de fuente y si el evento es un material grabado el código de inventario y código de material. La interfaz permite modificar la fuente del evento cambiando de DISCO a FUENTE EXTERNA y viceversa, además contiene un botón que se habilita si la fuente es DISCO el cual permite acceder a la interfaz de búsqueda de material para que el usuario pueda seleccionar otro video para su reproducción.

La interfaz modificar evento carga una copia de la estructura de datos que contiene la información del evento a modificar en memoria antes de mostrarse para no tener problemas de exclusión mutua con el thread de reproducción de la guía. Una vez que el usuario ha realizado los cambios notifica a la interfaz principal los mismos y esta última con la ayuda de semáforos accede a la estructura general en memoria que contiene la guía ubicando el registro y lo modifica guardando los nuevos datos. Una vez efectuada la operación el registro se actualiza la interfaz gráfica principal para retroalimentar al usuario marcando además el registro como modificado en la columna de estado para que el usuario conozca en qué eventos ha modificado en la guía de programación.

La interfaz principal solamente permite acceder a la modificación de eventos que aún no han transmitidos, no se pueden modificar eventos que ya han salido al aire o el evento actual que ya está siendo emitido.

En el caso de materiales grabados la interfaz principal se encarga automáticamente de verificar si el archivo de video ya se tiene en el disco y si no es así solicitar al thread de descargas de video que lo obtenga desde el servidor de almacenamiento mediante FTP.

La interfaz modificar evento también sirve para agregar un evento nuevo a la guía, si se visualiza para este fin la diferencia es que el título de la ventana cambia a Agregar evento

y no carga los datos de un evento elegido sino que el usuario debe seleccionar la fuente, si la fuente de un evento a agregar es de disco entonces tiene que utilizar el botón de buscar video para acceder a la interfaz de búsqueda de material grabado. Para visualizar la ventana en el modo de Agregar evento el usuario tiene a su disposición varios botones en la interfaz gráfica principal como lo son:

- Agregar un evento al final de la guía
- Agregar un evento en la posición seleccionada y recorrer los demás hacia adelante incluyendo el seleccionado.
- Agregar un evento después de la posición seleccionada dejando el evento seleccionado en su posición y agregando el evento nuevo debajo del mismo.

Otra opción para acceder a la interfaz para agregar eventos son opciones de un menú contextual sobre la guía de programación, aunque estas solo disponen de agregar un evento en la posición seleccionada y agregar un evento después de la posición seleccionada. La interfaz de agregar evento al mostrarse crea una nueva estructura de datos con la información del evento a agregar y al cerrar la interfaz secundaria la interfaz principal se encarga de insertar el nuevo registro en la lista que contiene la guía de programación. La operación de inserción en la estructura global de la guía también se apoya en el uso de semáforos de secciones críticas para no tener problemas de exclusión mutua que deriven en un fallo grave.

Interfaz búsqueda de material grabado

La interfaz secundaria búsqueda de material grabado permite al usuario buscar un video en los servidores de almacenamiento para agregarlo o realizar una modificación en la guía de programación. En esta interfaz gráfica el usuario tiene varios filtros a su disposición para buscar el material en la base de datos como son:

- Búsqueda por código de inventario
- Búsqueda por código de material
- Búsqueda por nombre de material
- Búsqueda en el disco duro local

En estos filtros de búsqueda el usuario debe colocar la información completa o parcial, el Reproductor de Máster se encarga de buscar en la base de datos el material que contenga el dato proporcionado, esta búsqueda se realiza en la tabla TVP_MAT_TRANS. La última opción mencionada permite seleccionar un material directamente de la ruta donde la aplicación almacena los videos en el disco duro local, en este caso el nombre del archivo se interpreta como el código de material para todas las operaciones internas del Reproductor de Máster donde se ocupe esta información y se busca en la base de datos para determinar todos los demás datos, sin embargo es la opción menos utilizada ya que es más sencillo para el usuario utilizar alguno de los identificadores disponibles en la base de datos.

El usuario en esta interfaz gráfica tiene a su disposición un botón para ejecutar la búsqueda en la base de datos del sistema utilizando la información ingresada de acuerdo al tipo de filtro seleccionado, al finalizar la consulta muestra los resultados que coinciden con la información proporcionada en una tabla de resultados que incluye datos de cada registro como lo es código de inventario, código de material, nombre del material y duración. El usuario entonces debe seleccionar el que necesita de la lista y finalmente dar clic en el botón de la interfaz gráfica “Cargar Video”, en ese momento la ventana secundaria se cierra y se regresa a la interfaz de Agregar evento o Modificar evento cargando los datos del video seleccionado en la interfaz.

La ruta local para el almacenamiento de los videos se encuentra definida en la configuración del Reproductor de Máster y puede dirigirse a cualquier ubicación o incluso un disco diferente de donde está instalada la aplicación, el cual generalmente se utiliza solamente para el almacenamiento de los videos.

Como en cualquier búsqueda si la información es más concreta el número de resultados es menor y el usuario encuentra más fácilmente el registro que necesita, generalmente en operación se observó que el filtro más utilizado es el código de inventario, esto se debe a que es un dato que se maneja tanto en el sistema de tráfico como en el sistema de automatización y por lo tanto una gran cantidad de usuarios lo conoce, además al ser un identificador único para cada material la búsqueda arroja solamente un resultado.

Interfaz búsqueda de súpers o cintillos

La interfaz de búsqueda de súpers o cintillos es la última interfaz gráfica secundaria que el operador tiene a su disposición, su funcionamiento es similar a la interfaz de búsqueda de material grabado con la diferencia que solamente se pueden buscar súpers o cintillos a través de esta ventana.

Los filtros con los cuales cuenta el usuario incluyen la búsqueda por código de inventario, código de material o un material en el disco duro.

Los súpers o cintillos a diferencia de los archivos que contienen material grabado para bloques de programa o comerciales se encuentran en formato mov y en una ubicación dentro del disco duro del equipo distinta a los demás materiales grabados.

El operador puede acceder a esta interfaz secundaria presionando el botón “Buscar Pleca” en la interfaz principal, está diseñada para permitir el disparo manual de súpers.

Componentes internos del reproductor de máster

En este punto del informe se han expuesto los elementos que conforman gráficamente la interfaz principal del usuario y las acciones que se pueden realizar en cada control de la misma, sin embargo el Reproductor de Máster cuenta con varios elementos internos y componentes que llevan a cabo ciertas tareas específicas. Como se observa en el

diagrama a bloques de la aplicación algunos son pipes para el intercambio de datos con el servicio de interfaz con el hardware, otros son objetos que permiten cierta funcionalidad y algunos más son threads de ejecución. Varios de estos elementos internos son creados y se comunican directamente con la interfaz gráfica principal y otros no, sino que son dependientes de otros objetos y elementos del Reproductor de Máster.

En las siguientes secciones abordaré la explicación de cada uno de estos elementos internos y cómo es que funcionan en conjunto.

Pipe de comandos

La interfaz gráfica principal, y por lo tanto el thread principal de ejecución, contiene el extremo cliente del pipe de datos dedicado al envío de comandos. Esto se debe a que los paquetes de los comandos son solamente de 32 bytes y por lo tanto las operaciones de envío tardan un corto tiempo en ser realizadas por lo que no interfieren con otras actividades de la interfaz gráfica principal, además es el thread principal de ejecución en esta aplicación y se encarga de recibir todos los mensajes de notificación y sincronización por parte de los otros threads de ejecución. En el thread principal se conoce el estado global de todos los componentes de la aplicación y es ideal que desde este componente se realice el envío de los comandos a ejecutar en el servicio de interfaz con el hardware mediante el pipe de comandos.

La interfaz gráfica principal además se encarga de revisar el estado de la conexión del pipe de comandos con el extremo servidor del pipe en el servicio de interfaz con el hardware. La verificación se hace cada 2 segundos, esto se apoya en un componente del entorno de programación utilizado tipo timer, cada que este timer se ejecuta se revisa la conexión del pipe y se notifica al usuario acerca del estado mediante el indicador correspondiente en la interfaz gráfica principal que se ilumina en color verde si no hay ningún problema y rojo si hay algún problema con la conexión.

El momento en el cual los comandos se envían se expondrá más adelante como parte de la explicación de la interacción entre los elementos que componen la aplicación.

Objeto conexión de BD

El Reproductor de Máster ocupa una clase desarrollada en C++ dentro del área para gestionar las conexiones al servidor de base de datos y forma parte de nuestras herramientas de desarrollo. La clase de conexión a la base de datos permite establecer conexiones con servidores de base de datos Oracle a través del uso de la interfaz de desarrollo del cliente de Oracle OCI (Oracle Call Interface).

En ejecución el objeto de base de datos instanciado recibe la consulta a ejecutar en lenguaje SQL en forma de cadena. También recibe un apuntador a estructura de datos tipo lista donde devuelve los resultados de la búsqueda.

La clase maneja también el proceso de conexión y desconexión del servidor de base de datos, reintentos de conexión y el análisis de valores de retorno de las operaciones enviadas al servidor de base de datos que permiten determinar la necesidad de intentos de reconexión en automático.

El Reproductor de Máster al iniciar la ejecución crea una instancia de la clase de conexión de base de datos y establece una conexión con el servidor de base de datos cuyo alias, nombre de usuario y password se encuentran definidos en el archivo de configuración de la aplicación. Un alias de una base de datos de Oracle define en el cliente un nombre mediante el cual se conoce la dirección IP del servidor, puerto de conexión y el nombre de la base de datos a la cual se requiere conectar la aplicación además de otros parámetros configurables. En un cliente de base de datos Oracle es posible definir estos alias en un archivo especial llamado tnsnames el cual utiliza el cliente de base de datos para buscar los alias requeridos por las aplicaciones y determinar los datos del servidor al cual requiere conectarse, a esto se le conoce como resolver el alias de la base de datos.

Dentro del Reproductor de Máster todos los componentes que requieren realizar consultas a la base de datos ocupan este objeto común de la aplicación, el cual está protegido por semáforos que permiten enfrentar los problemas de exclusión mutua que podría generar el acceso de dos componentes al mismo tiempo a este objeto.

Al cerrar el Reproductor de Máster se cierra la conexión al servidor de base de datos Oracle para desocupar los recursos del servidor. El Reproductor de Máster además fue diseñado para poder utilizar un servidor de base de datos de respaldo en caso de que la conexión con el servidor principal de base de datos no se pueda establecer. En ese esquema de trabajo las bases de datos deben estar configuradas de tal manera que los datos se repliquen o bien todas las aplicaciones que realizan inserciones en la base de datos y actualizaciones deben hacerlo en ambos servidores. En el despliegue del sistema de automatización en el proyecto correspondiente a este informe esta característica no fue activada, pero fue desarrollada para soportar otros escenarios de trabajo.

El indicador del estado de conexión con el servidor de base de datos presente en la interfaz gráfica principal basa su estado en el resultado de las operaciones que reporta este objeto. La interfaz principal también cuenta con un botón que tiene la leyenda “Conecta BD” el cual sirve para intentar una re conexión con el servidor de base de datos inmediatamente, fue diseñado para utilizarse tras realizar un mantenimiento en los servidores de base de datos o si se presenta una falla que mantiene el servidor de base de datos sin disponibilidad por largo tiempo, una vez que el servidor ha sido reactivado se utiliza el botón para reconectar la aplicación y se revisa el indicador en la interfaz principal para saber si la conexión se ha podido restablecer.

Objeto de envío de correos

El Reproductor de Máster se desarrolló con la capacidad de enviar alertas vía correo electrónico acerca de materiales grabados faltantes que no se encuentren en el subsistema de almacenamiento. La aplicación cuenta con una clase que permite realizar el envío de los correos electrónicos.

El envío de correos se realiza a través de la inserción de registros en una tabla en la base de datos llamada ENVIO_CORREO. En esta tabla cada registro tiene los datos necesarios para el envío de correos tales como remitente, destinatario, asunto, el cuerpo del correo, fecha de envío, una bandera de estado y un identificador de grupo de correo. Una aplicación independiente al sistema de automatización se encarga de revisar la tabla periódicamente para identificar los registros de los correos que aún están pendientes por enviar lo cual se indica en la bandera de estado, Si hay correos pendientes por enviar obtiene los datos de cada correo y lo envía mediante los servidores de correo corporativos utilizando el protocolo SMTP, posteriormente dependiendo del resultado del envío actualiza la bandera del estado en el registro de la tabla. La aplicación se desarrolló como parte de una solución distinta al sistema de automatización con la finalidad de proveer un servicio de envío de correo sencillo de utilizar en el desarrollo de otras aplicaciones.

La aplicación de envío de correos también es capaz de identificar grupos de destinatarios a través de otra tabla llamada GRUPOS_CORREO donde se tienen las direcciones de correo de los destinatarios agrupadas con un identificador. De esta manera los registros de cada correo también pueden tener un identificador de grupo y la aplicación de envío en ese caso obtiene todos los destinatarios para el grupo. El envío se realiza bajo este esquema no solamente a una dirección de correo destino sino a tantas como el grupo contenga en la base de datos.

El envío de correos desde la aplicación de Reproductor de Máster se reduce a insertar los registros de los correos a enviar y el identificador del grupo al que va dirigido. El identificador del grupo está definido en la configuración de la aplicación, el asunto del correo incluye siempre la leyenda "Notificación automática de máster" concatenada con el nombre del máster el cual también está definido en la configuración. El objeto de envío de correos tiene un método público para definir el cuerpo del mensaje a enviar. La interfaz principal es la única que utiliza este objeto como se explicará más adelante en el informe.

Aunque este objeto de la aplicación se utiliza solamente para el envío de notificaciones de materiales faltantes, está diseñado para el envío de cualquier tipo de notificación que se solicite en el futuro.

La inserción de los registros correspondientes a los correos se hace a través del uso del objeto de conexión con la base de datos. En cuanto a la aplicación de envío de correos realicé su desarrollo y la diseñé como un módulo que permite a otras aplicaciones enviar fácilmente correos electrónicos, está programada en lenguaje C++.

Objeto Guía de Programación

La guía de programación en el Reproductor de Máster se almacena en un objeto diseñado para guardar esta información. El objeto es una estructura de datos tipo lista donde cada elemento tiene la información de un evento en la guía

La información de cada evento es almacenada en una estructura de datos que cuenta con las siguientes propiedades:

- Nombre del evento
- Duración planeada en la guía en milisegundos
- Duración al aire en milisegundos
- Hora de inicio en milisegundos, si el evento no ha sido transmitido es aproximada con la suma de las duraciones de los eventos anteriores y es exacta cuando el evento ya salió al aire, la aplicación registra el momento exacto.
- Hora en la que terminó el evento en milisegundos, se determina una vez que el evento ha sido reproducido al aire
- Estado del evento, un valor numérico que indica si ya está preparado el material para salir al aire, está al aire, tiene algún problema o ya fue reproducido
- Tipo de evento: Indica si es un comercial, un programa, un evento agregado o un evento bloqueable por cuetone.
- Fuente del evento: Almacena un valor numérico que permite saber si es un evento con un material grabado o de fuente externa.
- Identificador de evento en el sistema tráfico: Identificador único de cada evento en la guía generado en el sistema de tráfico y es único para los eventos de la guía de una fecha determinada. Los valores se pueden repetir en la guía de otra fecha, se tiene que guardar para generar los reportes de transmisión ya que el sistema de tráfico se basa en este valor.
- Código de inventario: Solo para materiales grabados
- Código del material: Solo para materiales grabados
- Ruta completa del video en caso de ser un evento con fuente grabada
- Duración del archivo grabado
- Canales de audio del archivo
- Muestreo del audio del archivo grabado
- Identificador único de reproducción para control que es enviado al servicio de interfaz con el hardware
- Identificador del evento interno del reproductor de máster. Es un índice que se incrementa de 32 en 32 para etiquetar cada uno de los eventos al momento de cargar la guía de programación.
- Fecha de la guía a la que pertenece el evento
- Fecha en la que el evento salió al aire.
- Indicador de inserción de logo
- Indicador de inserción de cuetone en la señal, puede almacenar un valor que identifica si se debe insertar antes de finalizar el evento el cuetone de inicio o el de final de segmento bloqueable en la señal de salida.

Las guías de programación tienen la particularidad de contener la información acerca de la programación que generalmente se extiende de 5 de la mañana a las 5 de la mañana del día siguiente. Si bien duran 24 horas no van de media noche a media noche. Debido a esto las guías se nombran con base al día en el cual inician y el Reproductor de Máster para efectos de control debe almacenar de qué guía proviene cada evento así como la fecha real en la cual salió al aire porque los eventos de media noche a las cinco de la mañana del día siguiente son transmitidos en una fecha posterior al nombre de la guía y todo esto debe ser incluido en los reportes de transmisión.

Además el Reproductor de Máster debe operar ininterrumpidamente lo cual hace que sea necesario añadir la siguiente guía diariamente por la madrugada y también es necesario depurar automáticamente de la memoria los eventos pertenecientes a guías pasadas, proceso que se explicará más adelante en el proceso de carga de guías. Por lo anterior se requiere guardar la fecha de la guía a la cual pertenece cada uno de los eventos.

El identificador de evento en el sistema tráfico es un número único para cada evento dentro de la guía de cierta fecha, es un número de 8 dígitos generado en el sistema de tráfico. La importancia de este identificador es que en los reportes de transmisión el sistema de tráfico se basa en este dato para saber si el evento salió al aire o no así como la hora de inicio de la transmisión y su duración al aire.

El identificador del evento interno del Reproductor de Máster es un número que se asigna en la carga de la guía a cada evento y se incrementa de 32 en 32 para ser compatible con otros sistemas de automatización que ofrece el área como soluciones y con los cuales se podría interactuar. A los eventos agregados en la guía de programación por los usuarios operadores se les asigna el siguiente identificador con incremento de 32 respecto al máximo presente en la guía del día. Con base a este identificador el Reproductor de Máster es capaz de reconocer cada evento de la guía para cierta fecha en particular, diariamente al agregar los eventos de la siguiente guía se reinicia el índice a 32. Entre otras cosas se ocupa para saber si hay supers programados en cierto evento, el control de cuetones, el control de la inserción de logos y algunos eventos dependientes de la cuenta regresiva de los eventos. Es independiente al identificador de evento en el sistema tráfico aunque podría tener la misma utilidad, se decidió al diseñar el sistema de automatización que no se debían tener dependencias a indicadores externos en la medida de lo posible ya que pueden llegar a contener algún error.

La guía de programación también contiene cintillos o súpers relacionados con bloques de programa grabados, es decir, son súpers que se reproducen automáticamente después cierto tiempo a partir del inicio del bloque de programa. Esta característica formó parte de requerimientos posteriores a la implementación inicial del sistema y actualmente se encuentra automatizada. El soporte para estos cintillos automáticos incluyó la creación de una lista secundaria a la guía de programación principal y una estructura de datos reducida, respecto a los demás eventos, que guarda los datos pertenecientes a los súpers programados contando con las siguientes propiedades:

- Código de inventario

- Código de material
- Nombre del súper
- Estado del material
- Duración
- Fecha de la guía a la cual pertenece el evento
- Fecha real de aire
- Hora de inicio
- Hora de fin
- Identificador de evento en el sistema de tráfico
- Identificador interno del Reproductor de Máster del evento sobre el cual se debe reproducir el súper.
- Tiempo en milisegundos a partir del inicio del bloque grabado tras el cual se debe reproducir el súper.

Esta información es suficiente para almacenar cada evento de súper con reproducción automatizada, y podemos además observar uno de los usos del identificador interno del Reproductor de Máster para cada evento ya que a través de éste queda referenciada la reproducción automática de súper. La lista de súper automáticos es independiente a la lista de la guía de programación principal, cuando se realiza la carga de la guía del siguiente día también se añaden los súper de la nueva guía con reproducción automática esta lista secundaria y se lleva a cabo la depuración diaria de los eventos que ya no se utilizan de días pasados.

En el proceso de carga de guía la información proveniente del sistema de tráfico es interpretada y almacenada en memoria, cada evento se almacena en la lista que le corresponda según su naturaleza.

Objeto receptor multicast

En las secciones pasadas se ha expuesto el envío de información desde el Reproductor de Máster hacia el servicio de interfaz con el hardware así como su recepción en el servicio. El servicio de interfaz con el hardware genera información de retroalimentación para el Reproductor de Máster, esta información se envía a través de un protocolo de red mediante transmisión ip multicast.

En el servicio de interfaz con el hardware se tiene un objeto encargado de abrir un socket a bajo nivel con el API de Windows WiniNet. El socket se configura para soportar el protocolo de red IPv4 y a nivel de transporte utiliza datagramas sobre el protocolo UDP. El protocolo UDP no contiene verificación en el destino de la recepción de paquetes de información ni reintentos de envío, sin embargo es utilizado ampliamente en escenarios de comunicación que requieren rapidez sobre confiabilidad lo cual es muy práctico al trabajar con aplicaciones que manejan audio y video digital incluso en sistemas de streaming. Los datagramas en UDP generalmente deben ser pequeños y contienen información en un buffer de longitud fija.

En el servicio de interfaz con el hardware se define una dirección de multicast y un puerto a los cuales se deben conectar los clientes que requieren recibir la información de retroalimentación. El direccionamiento multicast está diseñado para enviar la información a un grupo de receptores interesados en los datos. En el RFC 3171 se define que las direcciones del segmento 224.X.X.X están reservadas para ser direcciones de tráfico multicast. Este rango de direcciones IP conforman la Clase D.

En la transmisión de información por multicast el emisor envía los datagramas a la dirección de multicast y los equipos de ruteo se encargan de hacer las copias necesarias para hacer llegar los paquetes a todos los receptores que se hayan registrado para recibir la información de la dirección multicast. Otra de las particularidades es que en este método de transmisión la información es enviada desde el emisor aún sin existir un cliente que realice una petición, por lo que en otras palabras podemos decir que el servidor es la entidad que inicia la comunicación.

En el caso del sistema de automatización es el servicio de interfaz con el hardware la entidad emisora de la información o bien servidor, el cual envía a la dirección de multicast paquetes de información con una longitud de 64 bytes en los cuales se encuentra almacenada toda la información referente al estado del servicio y las acciones que está realizando. En el caso de estar reproduciendo un contenido grabado la información se envía cada vez que un cuadro es enviado al hardware de salida, en caso de que el servicio se encuentre emitiendo señal de estudio la notificación se envía cada 400 milisegundos. Existen otras notificaciones inmediatas cuando ocurren eventos como el final de un video grabado, el inicio de un nuevo video grabado o el cambio de fuente.

El Reproductor de Máster es el receptor de la información, para lo cual utiliza un objeto dedicado a leer el tráfico de la dirección de multicast y puerto definidos en el servicio de interfaz con el hardware a través de la tarjeta de red del equipo donde se ejecuta. La dirección de multicast y el puerto están incluidos en el archivo de configuración del Reproductor de Máster. Al iniciar la ejecución el Reproductor de Máster se crea el objeto receptor de tráfico multicast abriendo el socket a bajo nivel con el API de Windows WiniNet, indicando el soporte para el protocolo IPv4 y se configura para la recepción de datagramas UDP en la capa de transporte. El socket además se registra para recibir el tráfico de red de la dirección de multicast a través del puerto definido, a partir de ese momento el objeto queda en espera del arribo de información para realizar las operaciones de lectura.

El objeto receptor de multicast también incorpora un mensaje de Windows que se utiliza para notificar al thread principal en cuanto recibe tráfico de la dirección de multicast. Al recibir la notificación el manejador de mensajes del thread principal tiene una función que interpreta la información recibida en cada paquete de 64 bytes y de esta forma conocer el estado del servicio de interfaz con el hardware.

En la apertura de los sockets necesarios para el envío de información se debe indicar la interfaz de red mediante la cual se enviarán los datos a la dirección de multicast, por otro lado en los sockets receptores se debe configurar la interfaz local de red mediante la cual

se van a leer los datos de la dirección de multicast. Dado que las dos aplicaciones se encuentran en el mismo equipo los sockets se configuran para el envío de la información a través de la interfaz de la dirección de loopback del equipo, la cual es ideal para este tipo de tráfico porque solo es necesaria transmitirla entre aplicaciones en el mismo equipo. De esta forma el tráfico de multicast emitido por el servicio no es enviado a ninguna red a la cual esté conectado el equipo.

La información enviada desde el servicio de interfaz con el hardware en los paquetes de 64 bytes incluye lo siguiente:

- Estado general del servicio: Indica el tipo de fuente que está reproduciendo, o si se encuentra en espera o en bypass.
- Estado del video en reproducción: Indica si está al aire un video grabado, si está ejecutando el inicio de un video o ha llegado al final de un material grabado.
- Identificador de reproducción del video actual: Es el identificador único de reproducción del video grabado generado en el Reproductor de Máster en el momento de iniciar la decodificación, se utiliza para reconocer el video que está en reproducción en el servicio.
- Cuadro actual del video grabado: Indica el número de cuadro actual del video grabado que el servicio está enviado a la salida, se utiliza principalmente para conocer el tiempo que ha transcurrido al aire del video actual.
- Hora de inicio del video: Hora en milisegundos en la cual el video inició su reproducción dentro del servicio.
- Identificador del video anterior: Identificador único de reproducción del video anterior que fue reproducido en el servicio.
- Nivel del buffer del hardware de salida.
- Nivel del buffer de audio y video en el servicio de interfaz con el hardware.

La mayor parte de esta información corresponde exactamente a los indicadores de estado del servicio de interfaz con el hardware incluidos en la interfaz gráfica principal del Reproductor de Máster.

La función que interpreta esta información en el Reproductor de Máster la utiliza para las siguientes acciones:

- Actualiza indicadores de los niveles de buffer del servicio de interfaz con el hardware
- Identifica el archivo grabado al aire actualizando la información del evento correspondiente en la guía de programación
- Identifica los cambios de archivos y registra la hora final de cada archivo para indicar la duración al aire en la guía de programación así como en los reportes de reproducción
- Actualiza las cuentas regresivas tomando en cuenta la duración del archivo actual de la estructura de datos correspondiente y los cuadros reproducidos reportados por el servicio.

- Actualiza la etiqueta de estado general del estado del servicio indicando si está un evento grabado al aire o una fuente externa.
- Si es necesario incluir cuetones en la señal de salida, mediante la notificación de los cuadros reproducidos es capaz de detectar cuántos segundos faltan para finalizar el evento actual y activar la inserción del cuetone.
- Inicia la reproducción automática de súpers tomando en cuenta el tiempo transcurrido del evento actual, esto solo se puede hacer en eventos de fuente grabada.

Thread carga de guía

El thread de carga de guía es un subproceso del Reproductor de Máster que fue diseñado para realizar la operación de lectura del archivo de texto que contiene la guía de programación, interpretación de la información y carga en memoria de las estructuras de datos necesarias para almacenar cada evento en el objeto guía de programación sin interrumpir las demás tareas del Reproductor de Máster.

El thread inicia su ejecución como parte de las tareas de inicialización del Reproductor de Máster y todo el tiempo se mantiene en ejecución esperando una instrucción enviada desde el proceso principal de ejecución para realizar la operación de carga de guía. Mientras no recibe la instrucción de operación de carga el thread permanece alternando entre la ejecución de una iteración de su ciclo revisando si ha recibido alguna instrucción y el estado de “dormido”. Los threads del Reproductor de Máster así como los del servicio de interfaz con el hardware fueron programados de tal manera que pueden ser despertados inmediatamente con una llamada asíncrona de procedimiento o APC (*asynchronous procedure call*).

La guía de programación se generaba en el sistema de tráfico, la salida del proceso era un archivo de texto plano con una serie de columnas y renglones, en el cual cada renglón es un evento de la guía y las columnas los datos de ese evento. Se definió que diariamente los usuarios “programadores de guía” deberían generar este archivo con ayuda del sistema de tráfico, el cual a su vez es depositado en una carpeta compartida al menos un día antes de su fecha de transmisión.

El reproductor de máster descarga el archivo de la guía diariamente de forma automática a la 1 de la madrugada de la carpeta compartida donde fue depositado mediante FTP. Una vez descargado solicita al thread de carga de la guía lea su contenido, la operación incluye la lectura del archivo, carga de cada evento en memoria y el los eventos se añaden a la lista que contiene guía de programación. Al finalizar el proceso de carga el thread además depura la información de los días pasados. La hora de descarga fue diseñada contemplando dar una ventana de tiempo en caso de que se presente algún problema o alguna cuestión operativa por la cual la guía no puede ser entregada, la hora de descarga automática marca el límite de tiempo para la entrega de la guía.

La descarga por FTP se efectúa en un thread dedicado para ello cuya operación se dispara desde el thread principal cuando llega la hora de la descarga. Al finalizar la

descarga el thread de FTP genera un aviso al thread principal, el cual a su vez manda la instrucción al thread de carga de guía indicando que debe cargar la guía descargada.

La operación para cargar la guía de programación incluye la validación de la fecha la cual se encuentra en el nombre del archivo acompañada de un identificador. El thread está programado para validar y cargar solamente la guía del día actual o el día siguiente por motivos de seguridad para evitar errores en el sistema. En la operación de carga el thread abre el archivo de texto e interpreta los campos necesarios, el formato del archivo deriva de la sintaxis de productos comerciales y es compatible con los mismos. El identificador para los videos grabados que se maneja en el sistema de tráfico es el código de inventario, el cual está presente en cada uno de los eventos que requieren reproducción de material grabado dentro del archivo que contiene la guía de programación.

Al leer cada uno de los eventos de la guía el thread va generando las estructuras de datos necesarias para almacenar los datos de cada evento en la memoria. En el archivo con la guía los eventos tienen atributos que identifican si son bloques de programa, comerciales o sùpers planeados para reproducción automática, el thread en el proceso de carga distingue entre los diferentes tipos de material y envía las estructuras creadas en memoria a la lista de almacenamiento adecuada, ya sea la guía de sùpers programados o bien a la lista que contiene los bloques de programa y comerciales de la guía de programación.

Para cada uno de los eventos con fuente de disco, es decir grabados, el thread se encarga de validar mediante consultas a la base de datos en la tabla TVP_MAT_TRANS utilizando código de inventario si el registro ya cuenta con material ingestado o no.

Si el material ya fue ingestado entonces obtiene su código de material y los datos correspondientes al nombre, duración, tamaño del archivo registrado y ubicación dentro de los espacios de almacenamiento, además revisa si el archivo ya está en disco de almacenamiento local del Reproductor de Máster. Si el material ya está en el disco duro del equipo el estado del evento se guarda como cargado y listo para ser transmitido, en caso contrario se marca en el estado que debe ser descargado posteriormente de los servidores de almacenamiento. Los materiales que aún no han sido ingestados guardan en su estado un valor que permite identificar la falta del material.

El estado de cada uno de los eventos se visualiza en la interfaz principal en la tabla que representa la guía de programación.

El nombre de los eventos con fuente externa proviene del archivo de texto de la guía de programación así como la duración planeada, sin embargo no es exacta ya que dependen de las duraciones que se vayan transmitiendo en vivo según el contenido.

El thread de carga de guía también se encarga de indexar cada uno de los eventos con un número que se incrementa de 32 en 32 para identificar los diferentes eventos en tiempo de ejecución. En cada evento también se guarda la fecha de la guía a la que pertenecen, otra de las acciones realiza este proceso es la asignación de los cuetones en los eventos correspondientes y además, por regla de operación definida por el cliente, en

cada uno de los eventos identificados desde la guía como bloque de programa se guarda la indicación de insertar el logo automáticamente en la estructura de datos del evento.

El thread reporta el progreso del proceso de carga del archivo a la interfaz principal cada que carga un grupo de 10 eventos con la finalidad de que la interfaz gráfica vaya actualizando la tabla con la guía de programación mostrando la información agregada.

Al terminar la operación de carga el thread envía una notificación al thread principal con lo cual concluye el proceso.

Thread ejecución de la guía

El thread de ejecución de guía es el subproceso del Reproductor de Máster que fue diseñado para controlar la ejecución de la reproducción en orden de todos los eventos presentes en la guía de programación. Este subproceso es creado como parte de las tareas iniciales del proceso principal y se mantiene en ejecución todo el tiempo alternando con estados en los cuales “duerme”.

El thread de ejecución de la guía recibe instrucciones desde el thread principal para cambiar el evento en reproducción cuando está por finalizar el actual, para iniciar la reproducción, detener la reproducción de la guía o bien prepararse para terminar la ejecución de la aplicación liberando todos los recursos asignados. El thread es capaz de enviar notificaciones al thread principal mediante mensajes del sistema operativo para que se ejecuten otras acciones que dependen de los parámetros del siguiente evento a reproducir en la guía de programación.

Al arrancar, el thread de ejecución de guía inicializa su estado y queda en espera de recibir la orden por parte del thread principal para empezar a ejecutar la guía, esta instrucción incluye el evento desde el cual el usuario selecciona iniciar la reproducción de la guía. La instrucción se genera en el thread principal dependiendo de las operaciones realizadas por el usuario en la interfaz principal con los controles que le permiten iniciar la reproducción.

En cuanto el thread recibe la instrucción de inicio de reproducción busca en la lista almacenada en memoria que contiene la guía de programación el evento en el cual se haya indicado debe iniciar la reproducción. Una vez encontrado el evento a reproducir en la guía de programación el thread analiza la información del evento a reproducir. Una de las propiedades que revisa es el estado del evento para saber si es posible reproducirlo, si es un evento de fuente grabada entonces el video debe estar disponible en el disco y el estado no debe contener algún valor que indique un error, pero si el evento es de fuente externa entonces se verifica que la fuente sea una válida. El thread también se encarga de verificar que los estados sean consistentes y solamente se tenga un evento al aire en todo momento. Para iniciar la reproducción de un evento el thread también verifica que sea nuevo, es decir no haya sido reproducido antes y no sea el que está en reproducción.

En este subproceso se extraen las propiedades de cada evento que determinan el comportamiento de la señal externa y estas son comunicadas al proceso principal mediante mensajes, a su vez el proceso principal realiza las operaciones correspondientes y de ser necesario envía los comandos requeridos al servicio de interfaz con el hardware de manera que la señal de salida contenga todos los elementos indicados en la guía de programación.

Las propiedades que se extraen de cada evento son las siguientes:

- Identificador del evento interno.
- Inserción automática del logo en el evento
- Si es necesario activar algún cuezone de inicio o fin de segmento bloqueable.

Una vez extraída la información el evento entra en estado de “PRECUE”, lo cual significa que está próximo a ser transmitido en el lenguaje de la industria de la televisión, la interfaz gráfica principal es actualizada para mostrar el cambio en el estado del evento próximo a reproducir.

El thread de ejecución de guía también se encarga de analizar la fuente de cada evento para ejecutar las acciones necesarias en cada caso. También es capaz de detectar si hay una transición de una fuente grabada a externa o al revés, y según sea el caso notifica al thread principal si debe activarse el hardware de entrada o desactivarse para que sea comunicado al servicio de interfaz con el hardware.

En el caso de eventos con fuente grabada el thread de ejecución de guía envía el archivo de video al thread encargado de la decodificación de videos, mientras esto sucede el thread queda esperando hasta que termina la decodificación del video. Al finalizar la decodificación del video avanza al siguiente evento lo cual se realiza retomando el proceso que acabo de describir desde el punto donde busca en la lista el siguiente evento a reproducir y analiza su estado para determinar si es posible reproducirlo.

Para los eventos con fuente externa el thread de ejecución de guía envía las notificaciones al proceso principal y queda esperando, las notificaciones enviadas al proceso principal son suficientes para que se lleven a cabo las operaciones que involucran el hardware de captura. Una vez que el proceso principal detecta el final del evento con fuente grabada actual se envían inmediatamente los comandos al servicio de interfaz con el hardware que activan el envío de señal externa a la salida. Si el servicio de interfaz con el hardware ya se encuentra en fuente externa entonces si es necesario cambia de entrada externa o permanece en la misma con un conteo de cuenta regresiva de 3 segundos. El thread de ejecución de la guía permanece esperando mientras está al aire la fuente externa porque al ser una producción en vivo no se tiene un tiempo exacto para enviar a corte comercial.

En los eventos en vivo los operadores del estudio y el operador de máster se definió que se comunicarían mediante un intercom, con un conteo de voz regresivo sincronizan el paso a corte comercial, por eso en el Reproductor de Máster la cuenta regresiva se

soportó para facilitar esta tarea. En el momento que el operador de máster decide pasar a corte comercial el thread principal envía la instrucción al thread de ejecución de guía para pasar al siguiente evento, en ese momento busca el siguiente evento en la lista de la guía de programación y se repite el proceso descrito.

Otra acción que puede realizar el thread de ejecución de guía es detener la lista de reproducción si así lo indica el usuario. Si se encuentra en un evento de fuente grabada interrumpe la decodificación y regresa al estado inicial que se tiene al arrancar el thread, quedando en espera de instrucciones. Por el contrario si es un evento de fuente externa recibe una notificación del thread principal tras lo cual queda en espera de una nueva instrucción de inicio de reproducción.

La última acción que puede ejecutar el thread de ejecución de guías es la tarea de finalizar la operación que se encuentre realizando, lo que incluye interrumpir el proceso de decodificación, liberar los recursos y terminar su propia ejecución. Esta acción se ejecuta como parte del proceso de finalización al cerrar el Reproductor de Máster.

Thread de decodificación de videos

El thread de decodificación de videos es el subproceso del Reproductor de Máster desarrollado para la decodificación de audio y video almacenados en los archivos de media. Este thread se mencionó en la sección anterior, si el thread de ejecución de guía detecta que el siguiente evento a reproducir es de fuente grabada y por lo tanto un archivo comunica el nombre del archivo al subproceso de decodificación de videos para que ejecute la decodificación del archivo. Los archivos de video se encuentran en una ubicación en el disco duro local definida en el archivo de configuración.

El thread de decodificación de videos integra las librerías basadas en libav que permiten realizar la decodificación de audio y video. La biblioteca libav incluye soporte para una gran cantidad de codificadores lo cual podría dar un soporte multiformato al Reproductor de Máster, sin embargo se diseñó solamente para operar con archivos encapsulados en mp4 de manera que no haya diferencia entre formatos que puedan retrasar operaciones de lectura y tener un desempeño diferente con cada tipo de formato.

El thread de decodificación cuenta con un ciclo que está en ejecución esperando a que le llegue la petición de decodificación de un archivo. Al llegar la petición el thread se dedica exclusivamente a la decodificación del archivo solicitando extrayendo cada cuadro en formato UYVY y los paquetes de muestras necesarios para acompañar cada cuadro. La decodificación de audio y video se realiza de manera independiente, es decir, es posible extraer audio sin extraer video y viceversa bajo ciertos parámetros de tolerancia. Lo anterior se debe a que dada la naturaleza del encapsulado, no necesariamente está el audio y el video correspondiente almacenado contiguamente en los datos del archivo. Las funciones del thread se encargan de extraer cada cuadro de video y enviarlo a un buffer dedicado para almacenarlo temporalmente, también se diseñó un buffer para almacenar temporalmente el audio que es extraído del archivo. Los datos enviados a los buffers de

almacenamiento temporales incluyen los encabezados que serán posteriormente enviados e interpretados en el servicio de interfaz con el hardware.

El reproductor de máster fue desarrollado con soporte para reproducir archivos que contienen video en resolución tanto HD como SD no importando la resolución de la señal de salida, para la decodificación se ocupa la mayor resolución posible acorde a la resolución de salida. En otras palabras, si la resolución de la señal de salida es HD entonces todos los videos se convierten a resolución HD no importando si su formato original es SD. La conversión se realiza en tiempo real al momento de decodificar el video con las funciones proporcionadas por la biblioteca libav. Si la resolución de salida es SD entonces todos los videos son convertidos a esta resolución con las funciones incluidas en la biblioteca libav, en este caso el proceso es el descrito anteriormente como central cut en el servicio de interfaz con el hardware.

Para la señal de salida en resolución HD, la conversión del cuadro de resolución SD a resolución HD se realiza escalando la porción central de la imagen en SD hasta lograr una resolución de 1440x1080 pixeles con lo cual la relación de aspecto se mantiene y no se deforma. La porción restante de la pantalla a lo ancho se reparte en dos bandas laterales para rellenarse, estas bandas reciben el nombre de pillar box como lo describe Davis (2008). Cada pillar box tiene un ancho de 240 pixeles y un alto de 1080 pixeles, estas bandas laterales pueden ser negras o en el caso del Reproductor de Máster un requerimiento posterior a su puesta en producción fue dar soporte para incluir gráficos personalizados en cada canal dentro de los pillar box. En el thread de decodificación se agregan estos gráficos combinando la imagen de los pillar box con la imagen escalada extraída del archivo cuando se realiza up-conversion de SD a HD, cada pillar box además es una imagen independiente, por lo que cada banda lateral puede tener su propio diseño.

El thread de decodificación genera notificaciones por cada cuadro de video extraído de los archivos así como cada paquete de audio, estas notificaciones son enviadas a la interfaz principal para que se pueda mostrar al usuario la cantidad de cuadros decodificados y los indicadores que reflejan el estado del proceso de decodificación.

La decodificación termina si se detectan alguna de las siguientes condiciones:

- Se ha decodificado todo el contenido del archivo
- Hay una diferencia de más de 1 segundo entre el audio y video decodificado, lo cual generaría un problema de sincronía entre audio y video por lo que la operación se aborta.
- Sucede un error fatal de lectura reportado por la librería libav.

Al terminar la decodificación del video, el archivo se cierra y el thread de decodificación envía una notificación al thread de reproducción de guía con la cual termina la espera de la decodificación del archivo y continúa el ciclo correspondiente para reproducir el siguiente evento. A su vez el thread de decodificación al terminar la tarea de decodificación queda a la espera de una nueva solicitud, para lo cual despierta periódicamente.

Buffer de video

El buffer de video está diseñado para almacenar temporalmente los cuadros de video extraídos de cada archivo del subproceso de decodificación en espera de ser enviados al servicio de interfaz con el hardware. Es una cola circular de estructuras de datos que permiten almacenar tanto el encabezado con las características del cuadro y las propiedades que se esperan en el servicio de interfaz con el hardware así como la información de la imagen en espacio de color UYVY.

Buffer de audio

El buffer de audio es una cola circular de estructuras de datos diseñada para almacenar los paquetes de muestras de audio extraídos de los archivos de video en formato PCM por el thread de decodificación así como los encabezados con las propiedades requeridas en el servicio de interfaz con el hardware. Los datos de audio se almacenan en este buffer temporalmente para su posterior envío al servicio de interfaz con el hardware.

El buffer de audio y el de video fueron diseñados para no interrumpir el proceso de decodificación por retrasos ocasionados en el proceso de envío de datos al servicio de interfaz con el hardware que incluye las tareas de escritura de datos en los pipes.

Thread de envío de video

El thread de envío de video es un subproceso del Reproductor de Máster diseñado para revisar si hay información en el buffer de video temporal donde están los cuadros decodificados de los videos y se encarga además del envío de los mismos mediante un pipe de datos al servicio de interfaz con el hardware. El extremo cliente del pipe de datos de video es creado en este thread, además el subproceso se encarga de revisar constantemente la conexión con el extremo servidor en el servicio de interfaz con el hardware y de ser necesario realizar alguna reconexión. De tal manera que no importa si el extremo servidor del pipe inicia antes o después que el Reproductor de Máster. El estado de la conexión del pipe se notifica a la interfaz principal para encender los indicadores correspondientes.

El thread de envío de video se encarga de notificar a la interfaz gráfica principal las tareas realizadas con mensajes, los cuales se interpretan y permiten dar a conocer el nivel del buffer temporal de decodificación al usuario.

Thread de envío de audio

El thread de envío de audio es el subproceso desarrollado para revisar la información del buffer temporal de audio y si hay paquetes de audio almacenados en el mismo los envía al servicio de interfaz con el hardware. El extremo cliente del pipe de datos de audio es creado en este thread, y al igual que el anterior, las tareas de revisión del estado de la conexión del pipe se realizan en este subproceso así como las reconexiones en caso de ser necesarias.

El thread de envío de audio también notifica a la interfaz principal acerca del estado de la conexión del pipe para activar los indicadores correspondientes así como los contadores que nos permiten saber si el audio se está enviando al servicio de interfaz con el hardware y cuál es el nivel del buffer temporal de audio.

Thread de descarga de videos

El thread de descarga de videos es un subproceso diseñado para descargar los archivos con material grabado desde los servidores de almacenamiento al equipo donde se ejecuta el Reproductor de Máster. El subproceso maneja una lista de peticiones que son solicitadas por el proceso principal, cada petición contiene el nombre del archivo que debe descargar así como la ruta en cada uno de los servidores de almacenamiento determinada a través de las consultas a la base de datos en las que se revisa si ya tiene material ingestado o no.

En el momento que recibe la petición el subproceso verifica si el archivo a descargar no se encuentra ya en la lista de peticiones para evitar la descarga repetida del mismo archivo desde los servidores de almacenamiento.

Al inicio de la ejecución el Reproductor de Máster monta una unidad compartida de cada servidor de almacenamiento con permisos de solo lectura. Los datos para montar la unidad se encuentran en el archivo de configuración como lo son usuario, password, ruta del compartido en red y letra de unidad a asignar. En el servidor de almacenamiento los directorios compartidos contienen enlaces simbólicos a los directorios que alojan los videos en cada disco del servidor. De esta forma al montar una sola ubicación se tiene acceso a todos los discos de almacenamiento del servidor. El proceso de copia se simplifica entonces a la copia de archivo de una unidad remota a la ubicación local. La ruta remota del archivo es determinada por la letra de la unidad asignada al montar la unidad compartida, el nombre del enlace simbólico en el cual se encuentra el archivo y el nombre del archivo que se requiere descargar.

El subproceso se apoya para realizar la copia de los archivos en el uso de las funciones asíncronas de copia de archivos presentes en el API de Windows. Las de funciones de copia de archivos permiten definir un callback el cual se llama para reportar el avance del proceso, mediante el uso de este callback se generan notificaciones a la interfaz principal para reflejar el avance en la barra que muestra el porcentaje de descarga del archivo.

Otros de los aspectos considerados en el diseño y pruebas del Reproductor de Máster fue la velocidad a la cual se realiza la copia de los archivos, porque hay que tomar en cuenta que además de las operaciones de escritura al disco realizadas por este subproceso de descarga también se tienen las operaciones de lectura realizadas por el proceso de decodificación. De hecho la velocidad de descarga se tuvo que ajustar para no interferir con las operaciones de decodificación y es que si bien son dos subprocesos diferentes a nivel de ejecución el recurso al que acceden es el mismo, es decir el disco duro. La operación de decodificación no se puede detener, de lo contrario afecta la señal de salida,

es por esto que se decidió limitar la velocidad de descarga para no interferir con las operaciones de lectura.

Las peticiones de descarga son atendidas de forma secuencial y solamente se realiza una a la vez. Los archivos se descargan a la ruta de almacenamiento definida en el archivo de configuración conservando el nombre que tienen en el subsistema de almacenamiento, el cual es el código de material. Una vez terminada la descarga se notifica a la interfaz principal donde se verifica que el video haya sido correctamente descargado, se recargan las propiedades del mismo incluyendo su duración y se actualizan todos los eventos con fuente local que vayan a necesitar el material recién descargado marcando su estado como cargado y listo para la transmisión.

Thread decodificación de sùpers

El thread de decodificación de sùpers es un subproceso del reproductor desarrollado para extraer las imágenes con transparencia de los contenidos tipo sùper almacenados en formato MOV. Este subproceso también se apoya en la integración de la librería libav para la decodificación de los videos. La decodificación también se realiza a la máxima resolución en la que se configura la señal de salida del módulo de reproducción, en este caso no es necesario incluir pillar box si se está realizando up-conversion.

A diferencia del proceso anterior de decodificación este subproceso está dedicado únicamente a la extracción de imágenes almacenadas de los archivos de sùpers y tiene que hacerse muchas de las veces al mismo tiempo que la decodificación de contenidos de programa grabados porque hay que recordar que son imágenes que se superponen en la pantalla. Los sùpers no contienen audio, solamente se decodifican imágenes con canal de transparencia en formato RGB Alpha en el que la profundidad en cada componente es de 8 bits. En este formato en cada pixel tiene 4 componentes, entonces cada pixel requiere 4 bytes para ser almacenado.

El subproceso se crea al iniciar la ejecución del Máster de Vivos y queda esperando una petición de decodificación. Antes de generar la petición, la interfaz principal se encarga de configurar en el thread el nombre y ruta del archivo con el sùper. Posteriormente en el momento adecuado la interfaz principal envía la instrucción para iniciar la decodificación del archivo, la petición es atendida por este subproceso y se encarga a partir de ese momento de la decodificación completa del archivo.

Los cuadros extraídos de los archivos del sùper se mandan a un buffer de plecas que los almacena temporalmente para su posterior envío al servicio de interfaz con el software mediante el pipe de datos dedicado para ese fin. Cada cuadro contiene además el encabezado con la información necesaria en el servicio de interfaz con el hardware que incluye las propiedades de la imagen.

Una vez terminada la decodificación del sùper el thread queda a la espera de otra petición.

Buffer de placas

Es una cola circular de estructuras de datos que fueron diseñadas para almacenar la información de los cuadros decodificados de los contenidos tipo súper, cada una de las imágenes se acompaña de un encabezado con sus propiedades. El buffer de placas es capaz de almacenar hasta un segundo de imágenes sin desbordarse, el thread encargado del envío debe obtener los datos y enviarlos lo más rápido posible al servicio de interfaz con el hardware para que esto no llegue a suceder.

Thread de envío de placas

El thread de envío de placas se encarga de obtener los datos disponibles en el buffer temporal de placas y enviarlos al servicio de interfaz con el hardware a través del pipe de datos dedicado para los súper. Al igual que los otros subprocesos de envío, el thread de placas se encarga de gestionar la conexión con el pipe de súper creando el extremo cliente del pipe de placas, realiza las operaciones de validación de la conexión y las reconexiones en caso de ser requeridas.

Este subproceso lanza notificaciones a la interfaz principal con el estado de la conexión del pipe de placas, lo que permite al proceso principal activar el indicador correspondiente en la interfaz gráfica principal con base en el estado reportado.

Socket de sincronización

El paso de una fuente externa a corte comercial o material grabado requiere asistencia del operador. También cabe recordar que se solicitaron para cada canal un equipo de reproducción principal y uno de respaldo, de manera que el cambio a corte comercial requeriría la manipulación de ambos equipos en simultáneo por parte del operador. Para evitar esto y que ambos equipos avancen al mismo tiempo se creó un mecanismo de sincronización basado en el envío de un buffer de datos mediante el protocolo de transporte TCP. El equipo de respaldo abre un socket en modo servidor para escuchar el tráfico mediante un puerto definido en su archivo de configuración. A su vez el equipo principal de reproducción abre un socket en modo cliente para conectarse al equipo de respaldo mediante el puerto y dirección IP definidos en su archivo de configuración. De esta forma cuando el operador acciona el cambio de evento en el equipo principal además de realizar las operaciones necesarias envía la instrucción de pasar al siguiente evento al equipo de respaldo mediante la red.

Inicialmente el protocolo de sincronización fue creado solamente para sincronizar los cambios de fuente, sin embargo en la actualidad también se ha soportado sincronizar desde el equipo principal operaciones de cambios en la guía tales como:

- Agregar un evento
- Borrar un evento
- Modificar evento

En cada caso se envía mediante el socket la información necesaria para que el equipo de respaldo realice en automático las mismas operaciones que el usuario está haciendo en el equipo principal. El control remoto en dirección del equipo de respaldo al principal no está definido ya que siempre se utiliza el equipo principal para la transmisión a menos que tenga un fallo fatal o de hardware. Si una falla se presenta en el reproductor principal, cuando se restablece es mejor copiar el archivo temporal con la guía del equipo respaldo e iniciar la operación cargando dicha guía con todos los cambios efectuados.

Con esto termina la exposición de todos los componentes que conforman el Reproductor de Máster, en la siguiente sección expondré el ciclo general de ejecución así como las interacciones entre cada uno de los componentes, aunque varias de ellas especialmente entre los pipes, buffer de datos temporales, threads de envío y de decodificación ya han sido abordadas.

Ciclo de ejecución del reproductor de máster

El reproductor de máster es una aplicación que fue diseñada para operar de forma autónoma la mayor parte del tiempo con excepción de los cambios entre fuente externa y grabado debido a la naturaleza de las producciones televisivas en vivo, así como la carga inicial de la guía y la elección del punto de inicio sobre la misma para la reproducción. Como ya se expuso otra de las operaciones no automatizadas es la introducción de cambios en la guía de programación, sin embargo la mayor parte de los canales para los cuales se diseñó este proyecto solamente incluyen materiales grabados en su programación, los cuales entregan con anticipación y los cambios en las guías son muy escasos. En ocasiones pueden pasar días o semanas sin que se solicite ningún cambio respecto a las guías de programación planeadas.

Una vez que el sistema ha iniciado la reproducción, en el caso de los canales con programación totalmente grabada el Reproductor de Máster se encarga de realizar todas las funciones necesarias incluyendo algunas tareas que en otros sistemas necesitan de un operador. En el caso de los canales que alternan señal en vivo y señal grabada se requiere un operador para realizar los ajustes de la guía y el paso a corte comercial.

El Reproductor de Máster inicia su ejecución creando todos los objetos internos para mostrar la interfaz principal de manera que le permita al operador cargar la guía de transmisión, para lo cual dispone de los controles en la interfaz principal. El operador tiene entonces dos opciones prácticamente para cargar la guía:

- Cargar una guía temporal
- Cargar una guía de texto proveniente del sistema de tráfico que ha sido depositada previamente en el repositorio de las guías

La guía temporal puede ser una guía previa almacenada que se generó como parte de la ejecución anterior del Reproductor de Máster o bien extraída de otro equipo de reproducción.

La guía de texto debe ser descargada con el control correspondiente en la interfaz principal previamente y utilizar la opción correspondiente en el menú de guías para cargarla en el Reproductor de Máster.

En ambos casos la aplicación revisa que la guía que se desea cargar sea la correspondiente a la fecha actual o del siguiente día. Esta es una medida de seguridad para evitar que el usuario se equivoque al cargar una guía de otra fecha. La validación se realiza con el nombre de los archivos, el cual incluye la fecha de la guía.

Una vez que el usuario selecciona la guía se comunica la ruta del archivo al thread de carga de guía, el cual se encarga de cargar la guía validando que los videos de cada uno de los eventos con fuente grabada se encuentren en el disco duro del equipo y para aquéllos que no encuentra en su disco duro genera las solicitudes necesarias al thread de descarga de videos. Además actualiza el estado de esos eventos para saber que la descarga de ese material está pendiente, lo cual se almacena en la estructura del evento correspondiente y se refleja en la guía de programación de la interfaz gráfica.

El avance carga de la guía se va mostrando paulatinamente al usuario en la interfaz gráfica, de hecho es posible que el usuario seleccione un punto desde el cual iniciar la reproducción o bien desde el primer evento apenas aparezca el evento necesario en la tabla de la interfaz gráfica que contiene la guía, es decir, puede iniciar la reproducción aunque la guía continúe siendo cargada.

La necesidad de contar con diferentes subprocesos en la aplicación se hace notar dado que debe realizar varias tareas simultáneamente y el usuario necesita iniciar la operación del Reproductor de Máster. El thread de carga de guía se encarga de continuar con el proceso de lectura de la guía y el thread de descarga de videos atiende las descargas pendientes de videos.

El usuario entonces debe elegir si la guía de programación se va reproducir desde el primer evento o si desea iniciar desde un evento en particular, para lo cual cuenta con el botón reproducir aquí y el botón de reproducir desde el inicio en la interfaz gráfica principal.

Al iniciar la reproducción la interfaz principal comunica al thread de ejecución de guía el punto a partir del cual el usuario ha seleccionado iniciar la reproducción. El thread de ejecución de guía atiende la petición buscando el evento en la guía desde donde se iniciará la reproducción y revisando las condiciones del mismo para iniciar, si fueran erróneas cambia al siguiente hasta encontrar uno que se pueda reproducir.

El thread de ejecución de la guía notifica el evento donde iniciará la reproducción a la interfaz gráfica principal, así como el identificador de reproducción interno del mismo y otros datos que incluyen las banderas con la información acerca de la inserción de logo automática activada, la presencia de un cuetone para el evento y el tipo de fuente del evento. La interfaz gráfica recibe la información e inicia el conteo regresivo de pre-roll de 3 segundos, también almacena las propiedades recibidas. Dependiendo de la fuente del

evento toma acciones para iniciar la reproducción, si la fuente del evento a reproducir es externa envía los comandos necesarios al servicio de interfaz con el hardware para preparar el dispositivo de entrada. Al término de la cuenta regresiva envía al servicio de interfaz con el hardware el comando para iniciar el envío de la señal de entrada a la salida, con lo cual al aire aparece la señal externa en pantalla.

En caso de un evento con fuente grabada el thread de ejecución de guía envía la solicitud de decodificación del archivo correspondiente al thread de decodificación, el cual una vez que ha decodificado 1 segundo de video notifica a la interfaz principal para que esta última registre que el inicio de la decodificación es exitoso. Mientras tanto el thread de decodificación junto con los buffers temporales y los threads de envío de audio y video se encargan de hacer llegar los cuadros del video y los paquetes de audio con las muestras al servicio de interfaz con el hardware. Al término de la cuenta regresiva la interfaz principal envía el comando de inicio de reproducción al servicio de interfaz con el hardware iniciando así la emisión del contenido del archivo de video en la señal de salida.

A partir del inicio de la reproducción el servicio de interfaz con el hardware reporta el estado de la salida al Reproductor de Máster mediante el mecanismo de comunicación multicast, el tráfico de red es escuchado por el receptor multicast en el Reproductor de Máster recibiendo notificaciones cuadro a cuadro en el caso de un evento grabado y cada medio segundo en el caso de eventos con fuente externa.

Con las notificaciones recibidas por multicast el Reproductor de Máster actualiza los elementos de la interfaz gráfica principal tales como contadores, reloj de cuenta regresiva si el material es grabado, indicadores de estado, etc. El Reproductor de Máster guarda la hora de inicio del evento al recibir la primera notificación de un cuadro enviado a la salida correspondiente al evento actual con precisión de milisegundos, misma que se utiliza para generar los reportes de transmisión.

Inmediatamente después de iniciar la reproducción de un evento grabado el thread principal revisa la lista de súper programados y si hay alguno planeado para el evento en reproducción verifica que el material del súper se encuentre en el disco duro, de ser así establece en el thread de decodificación de plecas ese archivo como objetivo de la decodificación. La búsqueda de la relación de un súper asignado para reproducción automática se hace con base al identificador interno de reproducción. Dado que la notificación recibida por parte del servicio de interfaz con el hardware es cuadro a cuadro, el Reproductor de Máster es capaz de indicar al thread de decodificación de súper el momento exacto correspondiente al tiempo programado en el cual debe iniciar la decodificación del súper, cabe recalcar que una vez que el servicio de interfaz con el hardware recibe 5 cuadros de un súper inicia su inserción en la señal de salida.

Una vez transcurridos los primeros 5 segundos del evento, si el evento tiene logo programado el Reproductor de Máster envía el comando para activar la inserción de logo en la señal de salida al servicio de interfaz con el hardware.

El manejo de la inserción de cuetones para los eventos en los que se tiene activa la misma se realiza de la siguiente manera dependiendo de la fuente:

- Fuente externa: Al solicitar el usuario el cambio al siguiente evento se manda el comando de inserción de cuetone correspondiente al servicio de interfaz con el hardware, por eso también existe el pre-roll de 3 segundos para garantizar la inserción de los cuetones con la anticipación requerida antes de cambiar al siguiente evento.
- Fuente grabada. Mediante las notificaciones por multicast el Reproductor de Máster determina el momento preciso en el que faltan 3 segundos para el término del evento y envía los comandos necesarios al servicio de interfaz con el hardware para la inserción del cuetone.

Las acciones que realiza el reproductor en el proceso principal al finalizar un evento son las siguientes, también dependen de la fuente:

- Al terminar un evento con fuente grabada detecta mediante el tráfico de multicast el cambio de archivo e introduce el registro del material al archivo de reporte de transmisión indicando la hora de inicio y la duración.
- Los eventos con fuente externa solamente pueden ser finalizados a petición del operador, en ese caso se esperan los 3 segundos de transición para introducir el registro al reporte de transmisión indicando hora de inicio y duración.

Los reportes de transmisión son archivos de texto con un formato similar al que contienen las guías de programación entregadas, sin embargo incluye hora real de inicio y duración, así como un elemento fundamental para el sistema de tráfico que es el identificador del evento en el sistema de tráfico.

El thread de ejecución de guía controla automáticamente la reproducción de los eventos mientras no exista un evento con fuente local, en cuyo caso el operador tiene que dar la instrucción de finalizar el mismo. Ya que existen los buffers en el hardware y en el software los eventos grabados terminan su decodificación con tiempo suficiente para que el thread de ejecución de guía identifique el siguiente evento a reproducir, realice su preparación y notifique al proceso principal el siguiente evento a reproducir. En el caso del final de los eventos con fuente externa la indicación de cambio por parte del usuario incluye el pre-roll de 3 segundos que es tiempo suficiente para que el thread de ejecución de guía identifique e inicie el procesamiento del siguiente evento.

Puede haber varias combinaciones de transición entre eventos dependiendo de la fuente de cada uno de los eventos:

- Fuente grabada a fuente grabada: El thread de ejecución de guía se encarga automáticamente al revisar el siguiente evento e iniciar su decodificación, comunicando además los atributos del siguiente evento a la interfaz principal.
- Fuente grabada a fuente externa: La interfaz principal debe enviar el comando al servicio de interfaz con el hardware para preparar el dispositivo de entrada y al

recibir la notificación del último cuadro del video manda el comando para que la señal de entrada se envíe al dispositivo de salida.

- Fuente externa a fuente grabada: Cuando el operador solicita el cambio de evento se notifica al thread de ejecución de guía el cambio, el cual envía el archivo al thread de decodificación. Tras una espera de 3 segundos la interfaz principal envía al servicio de interfaz con el hardware el comando para enviar a la salida los datos del video grabado en vez de la señal de fuente externa, mientras la espera transcurre el Reproductor de Máster envía audio y video suficientes a los buffers del servicio de interfaz con el hardware. En el momento que el video grabado ya se encuentra en reproducción se envían los comandos para desactivar el hardware de entrada y dejarlo listo para una próxima activación. Se envía además de ser necesaria la notificación de sincronización al equipo de respaldo.
- Fuente externa a fuente externa: Existe la posibilidad de tener dos eventos de fuente externa en la guía de programación en posiciones contiguas, para lo cual también el thread de ejecución de guía es notificado en cuanto el usuario solicita el cambio. El thread de ejecución de guía notifica en este caso al thread principal que sigue un evento de fuente local y el dispositivo que se desea utilizar. El thread principal revisa si el dispositivo requerido por el siguiente evento es el mismo, si es el caso solamente espera la cuenta de 3 segundos y da por terminado el evento actual sin realizar alguna acción con el hardware y registra el inicio del siguiente evento. En caso de que el dispositivo sea diferente envía los comandos necesarios al servicio de interfaz con el hardware para preparar el dispositivo requerido y tras los 3 segundos de espera activa el envío a la salida de la señal recibida por el nuevo dispositivo, posteriormente desactiva el dispositivo del anterior que ya no es requerido. Si es necesario se envía la notificación de cambio de evento al equipo de respaldo.

Inmediatamente después de realizar la transición se registra en cada caso en el reporte de transmisión el evento que acaba de terminar con los datos necesarios.

Los comandos recibidos mediante el socket de sincronización en los equipos de respaldo tienen el mismo tratamiento para realizar las transiciones entre eventos.

Una vez que finaliza un evento el thread de ejecución de guía se encarga de pasar al siguiente evento para reproducirlo y notificar al thread principal el tipo de fuente con lo cual se realizan las operaciones necesarias, entonces el ciclo se vuelve a repetir como fue descrito con cada evento.

El Reproductor de Máster se diseñó para automatizar una serie de tareas para lo cual algunos sistemas requieren un operador, las tareas automatizadas incluyen las siguientes:

- Revisión de materiales faltantes. Después de transcurrido 1 minuto y 30 segundos del inicio de cada evento identificado como bloque de programa el Reproductor de Máster verifica la presencia todos los archivos de eventos con fuente grabada en el disco duro. Si alguno no se encuentra en el disco duro revisa si se está en proceso de descarga o aún no han sido ingestados en el sistema. Si el video ha

sido ingestado pero no está siendo descargado solicita al thread de descarga de materiales que lo obtenga desde los servidores de almacenamiento.

- Notificaciones automáticas de material faltante: En el caso de que un material no esté en el sistema de automatización y falten 4 horas o menos para su transmisión envía un correo electrónico a los responsables, esta acción es complementaria al punto anterior y se reiteran notificaciones en cada bloque de programa hasta que el problema quede resuelto.
- Revisión de bloques de programa con material actualizado. Es posible que algún programa o bloque haya sido corregido y en consecuencia el material haya sido vuelto a ingestar, en ese caso el código del material será distinto. El Reproductor de Máster tras revisar en cada inicio de evento de bloque de programa los materiales faltantes, revisa también si existen cambios en los materiales de los bloques de programa. En caso de que haya cambios actualiza el registro correspondiente en la estructura de datos en memoria y solicita la descarga del nuevo video correspondiente al thread de descargas.
- Carga automática de guía: Diariamente a la 1 de la madrugada el thread principal del reproductor se encarga de descargar mediante FTP la guía que sigue a la fecha actual y ordenar su carga al thread de carga de guía, el cual añade el contenido del archivo a la guía actual. Al cargar la guía también verifica la existencia de los materiales en disco duro, si no se tienen en el equipo se solicita al thread de descarga de videos que realice la copia desde los servidores de almacenamiento.
- Entrega de reportes de transmisión: Al término de los eventos de la guía actual el Reproductor de Máster envía automáticamente el reporte de transmisión mediante FTP al repositorio de estos reportes que se encuentra en el mismo servidor de donde descarga las guías de transmisión generadas en el sistema de tráfico.
- Disparo de sùpers automático: El disparo automático de sùpers también se considera una virtualización de las funciones del operador, ya que anteriormente se tenía que realizar con los controles manuales sobre los bloques de programas grabados. En el caso de programas en vivo se debe hacer de forma manual debido a la naturaleza del contenido que proviene de un estudio.

Con la automatización de estas tareas el Reproductor de Máster desarrollado es capaz de reproducir la guía de programación de manera autónoma las 24 horas del día sin interrupciones, las únicas intervenciones de un usuario son para cambiar de evento en programas en vivo y realizar modificaciones a la guía de programación entregada.

El operador tiene a su disposición controles manuales para introducir logos, reproducir placas, programar cuetones, verificar videos y realizar operaciones sobre la guía de programación en las cuales puede modificar eventos, eliminarlos y agregarlos, también puede revisar algún video grabado en el visor de previo para asegurarse que el material es correcto. Dichos controles se incluyeron para cubrir cualquier necesidad no prevista cuando se crea la guía de programación planeada en el sistema de automatización, también pueden suceder noticias de último minuto que requieran la modificación de los eventos de la guía de programación.

El manejo de los eventos agregados o modificados incluye la modificación o creación de las estructuras de datos necesarios en las interfaces secundarias y después el thread principal aplica los cambios en el objeto de la guía de programación. El objeto de la guía de programación se encuentra protegido con semáforos de control de acceso mutuo exclusivo para que no se presenten problemas al realizar el acceso desde varios subprocesos a este objeto.

El Reproductor de Máster es la aplicación más compleja del sistema de automatización debido a las tareas que realiza y que es el componente responsable de la transmisión de la señal. Participé realizando desde el análisis de requerimientos, el diseño completo de la aplicación, su programación, pruebas y mantenimiento. La aplicación fue desarrollada totalmente en lenguaje C++, integrando librerías del API de Windows para copia de archivos, conexiones FTP, manejo de sockets, pipes de datos, manejo de interfaz gráfica y comunicación entre procesos. También se integraron librerías basadas en la biblioteca de funciones libav y parte de las librerías del reproductor VLC para la reproducción de previos.

El Reproductor de Máster es un proyecto en constante mejora, el tiempo de desarrollo inicial fue de 3 meses con las características básicas. Posteriormente han solicitado características nuevas lo largo de 4 años de operación las cuales incluyen el soporte de sùpers, la virtualización de las tareas del operador, la inserción de los pillar box con diseño individual para cada canal y la automatización de la reproducción de los sùpers.

3.8.3 Depurador del reproductor de máster

El depurador del reproductor de máster fue una herramienta diseñada para ser auxiliar del módulo de reproducción. Dado que los videos a reproducir para los eventos de la guía con fuentes grabadas se descargan al disco duro de los equipos en los que se ejecuta el Reproductor de Máster, eventualmente los discos duros se empiezan a llenar de materiales que no son necesarios diariamente en las guías, aunque estos equipos tienen la capacidad para guardar los videos necesarios hasta para 20 días de programación.

Si el material no se depura los discos duros de los reproductores de máster se saturarían impidiendo la descarga de nuevos materiales a transmitir. Para realizar la depuración de estos discos duros se creó la herramienta depurador de reproductor de máster.

El depurador puede ejecutarse en el equipo del reproductor o en otro equipo con los privilegios necesarios para conectarse mediante la red el equipo donde está el Reproductor de Máster y montar unidades desde este último. Si el depurador se ejecuta en otro equipo es necesario que se comparta en el equipo del Reproductor de Máster la unidad con los videos en red y al compartirla configurar los permisos de escritura bajo las credenciales de una cuenta de usuario definida.

Al ejecutar el depurador accede a la guía temporal del Reproductor Máster para la fecha actual, si se ejecuta en un equipo diferente esta es descargada mediante FTP. El depurador carga la información guardada en el archivo temporal que siempre incluye la programación del día actual y un día anterior debido a cómo realiza el thread de carga de guía la depuración de la guía en memoria en el Reproductor de Máster. Con la lectura de la guía temporal el depurador identifica todos los códigos de material diferentes presentes en la programación. A continuación hace un listado de todos los archivos almacenados en la unidad de videos del Reproductor de Máster. Para realizar la depuración se compara el listado de los archivos en la unidad de videos con el listado de los códigos diferentes a utilizar según la guía temporal, los videos que no se encuentren en la guía temporal son eliminados del disco duro liberando todo el espacio posible. Antes de realizar el borrado el depurador muestra al usuario una serie de mensajes que contienen el resumen del número de videos detectado, el número de videos diferentes en la guía de programación, cuántos videos se van a borrar, la cantidad de espacio a liberar así como otros detalles, con esta información el usuario puede confirmar que el borrado se efectuará sin ninguna afectación a la guía de transmisión que se está ejecutando en el Reproductor de Máster.

La depuración de los discos es una tarea manual que se definió para ser realizada cada dos semanas. Se establecieron los días martes para que así la depuración conserve los materiales desde el inicio de la semana.

Como trabajo futuro se planea integrar esta característica al Reproductor de Máster o bien automatizarla.

El depurador del reproductor de máster es una aplicación que desarrollé en lenguaje de programación C++ cuyo en 3 días de trabajo incluyendo las pruebas, es una herramienta auxiliar. Participé realizando su diseño, desarrollo, pruebas e instalación.

Capítulo 4. Resultados

El resultado del proyecto es un sistema de automatización de máster que genera la señal de 4 canales de televisión, el cual inició operaciones dentro de una empresa de televisión en septiembre de 2014 y se encuentra funcionando hasta la fecha de la entrega de este informe de actividades profesionales.

El sistema de automatización de máster implementado cuenta con los siguientes componentes principales:

- Un módulo de ingestión Web para producciones televisivas, el cual fue programado en PHP y ejecutando un servidor Apache, al cual se puede acceder desde las cabinas de edición con cualquier sistema operativo.
- Un módulo de ingestión Web para el envío de comercialización y bloques de programa al sistema de automatización de máster, programado en PHP y ejecutando en un servidor Apache.
- Aplicaciones automáticas para la conversión a formato MP4 para todo el material ingestado.
- Dos servidores de almacenamiento para los videos ingestados con redundancia.
- Base de datos relacional para almacenar la información relativa de los videos a transmitir, canales, producciones televisivas por canal, control de capítulos por producción televisiva y control de ingestión.
- Servidores de grabación de señales a diferir con redundancia. Capacidad para grabar 3 señales en formato HD simultáneas las 24 horas del día y almacenar hasta 4 días de material en alta calidad.
- Un módulo para extraer el material a diferir del sistema de grabación con dos instancias instaladas.
- Un módulo de ingestión manual para ingresar al sistema de automatización materiales a ser diferidos, bloques de programa con prioridad urgente, comercialización y materiales tipo súper. Existen 5 instancias de este módulo para atender cualquier eventualidad.
- Un Reproductor de Máster con redundancia para cada canal en operación. De los 4 canales con servicio, 3 de ellos manejan programación 100% grabada por lo que no requieren de operador salvo para realizar modificaciones a la guía planeada, y el cuarto maneja tanto señal grabada como en vivo para lo cual requiere un operador.
- Manuales y documentación de cada uno de los módulos del sistema.

El tiempo de desarrollo inicial del sistema fue de 5 meses y se contó con equipo de 3 desarrolladores a mi cargo para realizar tanto el análisis de requerimientos, desarrollo, programación y pruebas. Algunas de las características se fueron mejorando o algunas nuevas fueron solicitadas por el cliente durante los 3 años de operación del sistema.

La mayor parte del sistema se desarrolló en lenguaje de programación C++ apoyado con la integración librerías basadas en LIBAV para el manejo de la compresión y descompresión tanto de audio como de video. Los módulos de ingestión fueron desarrollados con tecnologías WEB para ser multiplataforma tal como lo solicitó el cliente del proyecto, se utilizó primordialmente lenguaje de programación PHP, JavaScript y se ejecuta en servidores WEB Apache dedicados al alojamiento de los módulos.

Las características generales del sistema de automatización y beneficios son los siguientes:

- La base de datos se aloja en los servidores Oracle propios de la empresa donde fue implementado, por lo tanto no fue necesaria inversión para este fin.
- Aprovechamiento del hardware especializado en adquisición y generación de señal de televisión ya adquirido en el área para otros proyectos, por lo que la inversión también fue mínima.
- Aprovechamiento de servidores y estaciones de trabajo ya disponibles con sistema operativo Windows 7, no se realizó inversión para compra de equipos nuevos.
- Interfaz con el sistema de tráfico ya adquirido, se redujo el tiempo de desarrollo con base en experiencia de colaboraciones anteriores. La inversión de recursos para el desarrollo de la interfaz también fue la menor posible así como el tiempo de capacitación de personal operativo, el sistema de tráfico ya era de amplio uso.
- Soporte multiformato en la ingestión, los archivos entregados por los editores son automáticamente transformados a un único formato para el sistema optimizando la calidad de la señal y el espacio de almacenamiento requerido en formato MP4.
- Módulos de ingestión multiplataforma accesibles desde las cabinas de edición. Los servidores de almacenamiento cuentan con redundancia
- Módulos para manejo de señales diferidas con alta calidad que permiten extraer las grabaciones a diferir en el mínimo de tiempo.
- Capacidad de manejo hasta para 6 señales externas en vivo en cada reproductor.
- Los reproductores de máster cuentan con redundancia
- Soporte para cuetones en la señal de salida.
- Soporte para súpers con disparo manual y soporte para reproducción automática sobre bloques de programa definidos en la guía de programación.
- Soporte de logo animado y logo estático
- Alto grado de automatización en la reproducción de la guía de programación.
- Envío automático de alertas sobre materiales faltantes.
- Generación y entrega automática de reportes de reproducción al terminar la ejecución de una guía de programación
- Desarrollo continuo de mejoras y nuevas características a la medida de las necesidades del cliente con tiempos de entrega ágiles y sin requerir grandes inversiones tanto monetarias como en tiempo.
- Las aplicaciones fueron diseñadas para ejecutar en equipos de bajo costo comparados contra las soluciones en el mercado que incluyen la adquisición de equipo dedicado, esto se traduce en un ahorro del 70% en la inversión de compra de hardware.

- Cuenta con un equipo de soporte altamente calificado que fue capacitado para resolver cualquier eventualidad por lo tanto no se invierte en esquemas de pago por el soporte técnico.
- La señal generada por el sistema de automatización es transmitida en el territorio nacional así como a Estados Unidos, Centroamérica, Sudamérica y Europa, lo cual constituye un gran ingreso económico.

Las ventajas de contar con un sistema de automatización desarrollado a la medida son muchas ya que reduce ampliamente los costos tanto de desarrollo e implementación, así como el soporte y licencias a ser adquiridas comparadas contra la compra de una solución comercial. Generalmente las soluciones comerciales tienen un esquema de renta de los equipos así como una cuota por el soporte técnico y si se requieren adecuaciones a la medida también tienen un costo. Por otro lado los tiempos de desarrollo para cubrir una necesidad pueden ser mucho más cortos comparados con productos comerciales que tienen oficinas de desarrollo en otros países o continentes, lo cual se traduce en el aumento de competitividad en la señal

El sistema de automatización de máster ha generado un nuevo paradigma en la empresa cliente debido a que, incluso en el momento de su creación, no se creía posible llegar a contar un sistema de este tipo totalmente automático para canales con programación grabada, lo cual se logró con este proyecto.

Algunos de los módulos de este proyecto y en especial el Reproductor de Máster fue aprovechado en televisoras locales para generar programación en vivo así como operadores de televisión en Centroamérica se interesaron en su adquisición, en esos casos el sistema de automatización constituye el control maestro de la señal. Lo anterior ha requerido la adaptación de ciertos módulos con la finalidad de tener una interfaz con otros equipos de almacenamiento que no sean necesariamente los servidores centrales descritos en este informe.

Finalmente el sistema de automatización ha resultado ser altamente confiable durante los 4 años que lleva en operación, se han tenido muy pocos incidentes con consecuencias al aire, la principal causa de estas incidencias se debe a problemas de edición del contenido. El sistema de automatización de máster cuenta con un plan de mantenimiento el cual es ejecutado por el equipo de soporte para prevenir cualquier problema. La confianza generada en el sistema de automatización ha hecho que se proyecten planes futuros para su mejora y adopción de uso en la generación de más señales.

Conclusiones

La industria de la televisión actualmente se apoya en gran medida en la tecnología que ofrece el procesamiento de la información en sistemas y equipos de cómputo tanto para almacenamiento y producción de materiales audiovisuales así como en la transmisión de la señal. Sin el soporte de las tecnologías disponibles sería imposible llevar a cabo las tareas necesarias bajo las restricciones de tiempo y calidad que se exigen. En la actualidad las transmisiones de televisión abierta y privada son totalmente digitales.

La competitividad de una empresa en la época moderna depende enormemente de los procesos que sean automatizados, lo cual representa beneficios en las operaciones tanto en el tiempo en el cual las acciones son ejecutadas, un menor margen de error humano y la inversión necesaria para llevar a cabo los procesos. Los resultados se esperan rápidamente y de forma oportuna para aprovechar oportunidades que pueden ser únicas.

La ingeniería mexicana es capaz de desarrollar sistemas que compiten con las mejores soluciones a nivel mundial de reconocidos fabricantes en el campo de la industria de la televisión. La creación de este tipo de tecnología permite ahorrar grandes cantidades de dinero y tiempo además de ser orgullosamente mexicana.

El trabajo en equipo tanto entre los integrantes de una misma área de desarrollo como con integrantes de otras áreas es fundamental para construir aplicaciones y soluciones adecuadas bajo las restricciones de tiempo e inversión destinada a un proyecto. Por ello se debe contar con las mejores prácticas posibles así como incentivar la comunicación y una buena actitud de colaboración entre los equipos de trabajo ya que persiguen un fin común. Incluso el trabajo en equipo puede incluir a integrantes de otras disciplinas que no necesariamente sean del campo de la ingeniería, entonces es muy importante que el ingeniero pueda plantear sus ideas de una forma clara y concisa así como contar con una capacidad de análisis y síntesis que le permita comprender las necesidades expuestas por el cliente que solicita el desarrollo un proyecto; de esta forma se pueden proponer incluso nuevas ideas que mejoren el producto aportando un valor agregado.

Conclusiones personales

A lo largo de mi trayectoria laboral he participado en diferentes proyectos de ingeniería relacionados con la industria de la televisión aportando los conocimientos adquiridos durante mi formación profesional. El trabajo diario en estos proyectos ha requerido tanto de la aplicación de dichos conocimientos como la investigación y actualización en nuevas metodologías y tecnologías para desempeñar las actividades de los cargos que he ido desempeñando en la organización donde laboro.

En particular en el momento de iniciar el proyecto para el desarrollo del sistema de automatización de máster, presentado en este informe, contaba ya 6 años de experiencia.

El desarrollo de este proyecto fue muy interesante ya que representó una de mis primeras oportunidades para tomar las decisiones fundamentales en el diseño de todo el sistema así como organizar un grupo de trabajo delegando la codificación de componentes.

La calidad de la solución y el conjunto de características incluidas están a la par de cualquier sistema de automatización comercial desarrollado por empresas dedicadas al desarrollo de tecnología para la industria de la televisión, lo cual es muy gratificante. La estabilidad que ha mostrado el sistema ha generado confianza entre los usuarios y las áreas ejecutivas, permitiendo así contar con un sistema de bajo costo para generar la señal de varios canales en una televisora. La señal resultante es emitida en México y en otros países, lo cual constituye una prueba de los resultados del proyecto así como la satisfacción de saber que el trabajo invertido en el desarrollo tiene impacto en muchas personas alrededor del mundo y es palpable.

Una de las limitantes en este proyecto fue que el hardware destinado para el mismo ya se había adquirido con anterioridad e incluso había sido utilizado en otros proyectos. Por lo tanto una de las directrices era hacer uso tanto de los equipos de cómputo disponibles, del hardware de captura y reproducción de la señal, como también la infraestructura de redes y bases de datos corporativa. Sin embargo, esta limitante no fue un obstáculo para el desarrollo del proyecto sino una oportunidad para hacer un sistema robusto y eficiente que utilizara los recursos disponibles minimizando la inversión.

La propuesta del manejo de información para los programas grabados y el control de números de capítulos es una de las grandes ventajas del proyecto, en este punto es importante recalcar que en muchos proyectos es posible proponer ideas que van más allá de los planteamientos iniciales del cliente buscando explotar áreas de oportunidad con la finalidad de mejorar sus procesos operativos apoyados por sistemas de cómputo que van más allá de sus expectativas.

El trabajo en equipo fue fundamental en el desarrollo de este proyecto tanto con ingenieros como profesionales en otras disciplinas, con lo cual corroboro la importancia que tiene para un ingeniero saber entablar este tipo de trabajo.

En este proyecto también observé la importancia de poder reutilizar componentes y conocimientos adquiridos anteriormente para cumplir con las fechas de entrega. También se comprobaron las ventajas que representan los diseños modulares para cualquier sistema las cuales son evidentes en el planteamiento de la solución del proyecto.

Los conocimientos adquiridos durante mi formación en materias como programación, fundamentalmente en lenguaje C, lenguajes de programación, procesamiento digital de imágenes, computación gráfica, diseño de algoritmos, estructuras de datos, sistemas operativos, arquitectura de microcomputadoras, procesamiento de señales, ingeniería de software, bases de datos e incluso redes fueron de vital importancia para la realización de este proyecto. Algunas de las ideas fueron exploradas de forma básica durante los trabajos académicos, pero sin duda alguna son las bases que utilicé para poder desarrollar los elementos del el sistema de automatización de máster y adquirir las habilidades necesarias para entender nuevas tecnologías de diversos tipos.

El sistema de automatización de máster es uno de los proyectos más completos y complejos en los cuales he colaborado, ha requerido una inversión de tiempo mayor comparativamente con otros proyectos pero el resultado final cumple con todas las características solicitadas, es muy satisfactorio.

Finalmente las características más atractivas de este proyecto son que se encuentra en constante desarrollo y aún tiene áreas de oportunidad relacionadas con la computación gráfica y la programación, las cuales son dos áreas de gran interés profesional para mí desde la formación académica. Lo anterior se debe a que los productos comerciales y las necesidades en cuanto a la transmisión en la industria de la televisión siempre están evolucionando para adoptar nuevas ideas en el procesamiento y presentación de gráficos. Aún queda trabajo por hacer para incluir más características presentes en otros productos así como requerimientos que pudieran haber en el futuro, para lo cual el sistema de automatización de máster que diseñé y desarrollé cuenta con una base sólida y confiable.

Glosario

ACL (Access Control List) o lista de control de acceso: Mecanismo para determinar los permisos de acceso a un recurso dependiendo del proceso que se realiza para acceder. En los equipos de red las ACL permiten controlar el tráfico y su objetivo es filtrarlo permitiendo o denegando el acceso a la red y equipos de acuerdo a alguna condición

Alpha compositing: En los gráficos por computadora, es el proceso de combinar una imagen con un fondo para crear la apariencia de una transparencia parcial o total. El canal alpha define qué grado las imágenes combinadas se funden.

API (Application Programming Interface) o Interfaz de programación de aplicaciones: Conjunto de subrutinas, funciones y procedimientos que ofrece alguna biblioteca para ser utilizado en otro software. Representa la capacidad de comunicación entre componentes de software. Uno de los principales propósitos es proporcionar un conjunto de funciones de uso general lo cual permite a un software interactuar con otro software.

ATS (Advanced Television Systems Committee): Organización internacional encargada del desarrollo de los estándares de televisión digital en Estados Unidos y Norteamérica. El grupo fue creado para reemplazar el sistema de televisión análoga NTSC. ATSC trabaja generando y coordinando los estándares pertenecientes a diferentes medios de comunicación enfocándose en la televisión digital, sistemas interactivos y las comunicaciones multimedia de banda ancha.

Banda base: “Se denomina banda base al conjunto de señales que no sufren ningún proceso de modulación a la salida de la fuente que las origina, es decir son transmitidas en su frecuencia original” (Textos Científicos, 2005). Este tipo de señales no se adaptan al medio por el que se van a transmitir ni son comprimidas. Ejemplos de señales en banda base es la señal obtenida a la salida de video de dispositivos como grabadores o reproductores de video y consolas de videojuego que son transmitidas a cortas distancias.

Bitrate o tasa de bits: Como señala Christensson (2006), en informática y telecomunicaciones es el término que define el número de bits que se transmiten por unidad de tiempo a través de un sistema de transmisión digital. Es la velocidad de transferencia de los datos o la cantidad de información almacenada o transmitida en cierto tiempo. Generalmente se mide en bits por segundo, kilobits por segundo (Kbps) o megabits por segundo (Mbps).

BlackBurst: También conocida como señal de sincronía bi-nivel. Es una señal análoga utilizada en telecomunicaciones para sincronizar dos señales de video en el mismo plano de tiempo. Consiste en una señal visualmente en negros. Se utiliza también en los estudios para sincronizar los diferentes equipos.

Botonera o video router: Switch electrónico diseñado para rutear señales desde múltiples fuentes como cámaras, computadoras o reproductores de video a otros dispositivos de monitoreo o procesamiento.

Branding: Anglicismo que se emplea en mercadotecnia para hacer referencia al proceso de construir una marca. Román (2016) menciona que se trata por lo tanto de la estrategia a seguir para la creación de los elementos diferenciadores de la identidad de una marca y que contribuyen a la construcción de una experiencia de marca distintiva y completa.

Bug: Error en el software que lleva a producir resultados no deseados.

C#: Lenguaje de programación orientado a objetos desarrollado por Microsoft como parte de la plataforma .NET.

C: Lenguaje de programación desarrollado originalmente por Dennis Ritchie entre 1969 y 1972 en los laboratorios Bell. Produce código muy eficiente y probablemente sea uno de los lenguajes más populares y de amplio uso para la programación estructurada, es un lenguaje de alto nivel compilado.

C++: Lenguaje de programación diseñado a mediados de los años 80 por Bjarne Stroustrup con la intención de extender el lenguaje de programación C con mecanismos que permitan la programación orientada a objetos además de la programación estructurada y procedural que ya contemplaba.

Carta de programación: Documento donde se detallan los horarios de los programas de un canal de televisión sin hacer mención a la estructura interna de comercialización o bloques de programa.

Closed caption: Proceso mediante el cual se muestra texto en un televisor para proporcionar información, generalmente el texto es una transcripción del audio del programa de televisión que pueda incluir descripciones de elementos no verbales. Fue planteado para favorecer a la población con problemas auditivos y es parte de regulaciones gubernamentales que exigen su presencia en la señal de televisión abierta.

Comercial: Material audiovisual que se presenta a la audiencia a través de la señal de televisión. Su duración típicamente se encuentra entre los 10 y los 60 segundos para promocionar algún producto, servicio o institución. Los contenidos con duración de un minuto y hasta los 5 minutos pasan a denominarse cápsulas. Cuando superan los 5 minutos entonces se denominan infomerciales y se asemejan a un programa televisivo.

Compresión: En computación, es la reducción del volumen de datos para representar determinada información en una menor cantidad de espacio. La compresión es un caso particular de codificación cuya característica principal es que el código resultante tiene un menor tamaño que el original. La compresión de datos se basa fundamentalmente en buscar repeticiones de series de datos para almacenar una vez el dato con el número de repeticiones que contiene la información apoyándose en algoritmos complejos. Existe compresión sin pérdida en la cual los datos antes de la compresión y después al

extraerlos son exactamente los mismos. Por otro lado existen algoritmos con pérdida que generalmente disminuyen más el tamaño necesario para almacenar la información pero también la calidad ya que una vez realizada la compresión no se puede obtener la señal original sino una muy semejante.

Crominancia: Información que contiene todo lo relacionado con el color de los objetos de una imagen. Las señales de crominancia se obtienen de las llamadas señales diferencia de color, estas son Rojo menos Luminancia , Azul menos Luminancia y Verde menos luminancia. De esas tres señales solo se necesitan dos ya que la tercera se puede obtener mediante la combinación de las anteriores.

CSS3 (Cascading StyleSheets) u Hojas de estilo en Cascada: Como indica W3C (2016) , es un lenguaje de diseño gráfico creado para definir la presentación de un documento estructurado escrito en un lenguaje de marcado, Se utiliza a menudo para establecer el diseño visual de los documentos web e interfaces de usuario escritas en HTML o XHTML.

Cuadros B o Bi-predicted frames: Imágenes que se generan a partir de la interpolación de otras dos de referencia. Las imágenes de referencia pueden ser de tipo I o P y debe ser una anterior en el tiempo y una posterior. Generalmente son las que requieren menor cantidad de información para ser almacenadas ya que son interpoladas.

Cuadros I o Intra frames: Imágenes cuya codificación depende de sí mismas. No tienen referencias a otras imágenes. Se utiliza la mayor cantidad de bits para codificarlas y se puede empezar a reproducir un video desde este punto sin problema ya que contiene toda la información necesaria para recuperar la imagen.

Cuadros P o Predicted frames: Imágenes que se reconstruyen por extrapolación agregando información a una imagen anterior en el tiempo, es decir se generan a partir de una referencia a una imagen I o P anterior. Para esto se utilizan estimaciones de movimiento a partir de las imágenes de referencia.

Cuetone: Mensaje compuesto por tonos de audio que se envía para ejecutar una acción en equipos remotos. Dentro de la transmisión de televisión los cuetones son tonos DTFM los cuales se escuchan como los tonos que existen en las líneas telefónicas donde tuvieron su origen.

dbFS: Abreviatura que significa “decibelos a escala completa” (decibels full scale). Representa una unidad de medida para niveles de amplitud en decibelios dentro de sistemas digitales que manejan una amplitud máxima. En esta escala de medida el máximo valor posible es 0.

DBLink: “Es una conexión entre dos servidores físicos de base de datos que permiten a un cliente acceder a ellas como una misma base de datos lógica” (Oracle, 2018).

Disco de Nipkow: Como menciona Palazzesi (2010), es un dispositivo inventado en 1884 por Paul Nipkow capaz de explorar una imagen. Era un disco metálico con una serie de

agujeros cuadrangulares dispuestos en espiral. Al hacerlo girar cada agujero recibía parte de la luz reflejada por la escena que estaba siendo adquirida, fue el primer dispositivo capaz de convertir una imagen en una serie de impulsos eléctricos.

Down-conversion: Proceso para convertir una imagen en resolución mayor, generalmente en formato HD, a una resolución menor, generalmente en formato SD. Puede utilizar diferentes métodos.

DTFM (Dual-tone multi-frequency signaling): Señalización basada en tonos de audio que se emplea para generar los cuetones. El mecanismo de DTFM fue originalmente creado por Bell Sytem en Estados Unidos, está estandarizado bajo la recomendación ITU-T Q23. La DTFM tiene tonos para representar los 10 dígitos del sistema decimal así como letras de la A a la D y dos caracteres especiales # y *. Los tonos se generan en base a la suma señales senoidales, una de estas señales tiene una frecuencia baja que está entre 697 y 941 Hz y la otra tiene una frecuencia denominada alta que puede ir de los 1209 Hz a los 1633 Hz, fueron diseñados para ser recuperados en el otro extremo del canal de comunicación sin ningún problema.

DVB (Digital Video Broadcasting): Consorcio compuesto por compañías líderes en la industria de la tecnología y televisión digital alrededor de mundo como fabricantes, desarrolladores de software, operadores de redes, emisoras y reguladoras, que se encarga del diseño de estándares técnicos abiertos para la emisión de televisión digital y otros servicios de transmisión. Originalmente las empresas involucradas eran europeas lo cual originó que los estándares creados fueran adoptados principalmente en Europa, aunque su uso se ha extendido a regiones de África, Asia, Oceanía y algunos países en Sudamérica.

Espacio de color: Sistema de interpretación del color, en otras palabras es una organización específica de los colores de una imagen o video. En conjunto con un modelo de color, que es un modelo matemático abstracto para representar colores en forma numérica, y los dispositivos que permiten las representación de colores contenidas en señales analógicas o representaciones digitales se transmiten y visualizan imágenes en ese tipo de dispositivos.

Formato video HD: Sistema de imagen en el cual la resolución profesional en televisión es de 1920 x 1080 pixeles.

Formato video SD: Sistema de imagen en el cual la resolución profesional en televisión es de 720 x 480 pixeles.

Formato video Ultra HD: Sistema de video el ultra alta definición que incluye resolución 4K que tiene 3840 x 2160 pixeles y también incluye la posibilidad de 8K con resolución de 7680 x 4320 pixeles. Este formato fue propuesto en Japón.

Frame o cuadro: Nombre que se le da a cada imagen que compone una secuencia de video.

Framerate: Tasa de cuadros por unidad de tiempo que se reproducen en un dispositivo de despliegue para lograr una animación fluida percibida por el ojo humano. Generalmente se mide en cuadros por segundo o fps. Para el caso de la televisión en México la tasa profesional es de 29.97 fps ya sea en formato HD o SD.

Framework: Marco de trabajo con un conjunto estandarizado de conceptos, prácticas y criterios. En el caso del desarrollo de software se tienen artefactos o módulos concretos de software que sirven de base para los proyectos realizados. Un framework puede incluir programas, bibliotecas y otras herramientas.

FTP (File Transfer Protocol): Protocolo de red para la transferencia de archivos que permite la conexión entre dos equipos para transferir archivos bajo una arquitectura cliente - servidor.

GOP (Group of Pictures) o Grupo de imágenes: Secuencia de cuadros tipo I, P y B que se generan al codificar video en estándares MPEG. Especifica el orden de este tipo de cuadros y cuántos de cada tipo incluye cada secuencia. El GOP siempre debe empezar con un cuadro I y no incluye cuadro de este tipo hasta el principio de un nuevo GOP, por lo cual podemos decir que solo incluyen un cuadro tipo I mientras los demás son tipo B o P.

Guía de programación: Secuencia de eventos a transmitir en una señal de televisión que incluye bloques de programa y cada uno de los comerciales a transmitir entre esos bloques de programa. Generalmente incluyen las horas de inicio, duraciones, descripciones y alguna forma de identificar cada uno de los eventos.

HTML (HyperText Markup Language): Como se indica en MDN Web Docs Mozilla (2017) es el estándar que define los elementos de construcción más básicos de una página web y se utiliza para crear y representar visualmente una página web. Es un estándar a cargo del World Wide Web Consortium o W3C.

Iconoscopio: Dispositivo que es la adaptación de un tubo de rayos catódicos con el objetivo de capturar imagen. El equipo inventor del dispositivo dio inicio a la televisión electrónica fue liderado por el científico ruso Vladimir Zworykin para la compañía RCA en 1934. En el modelo original la imagen ingresa a través de una serie de lentes impactando las células fotoeléctricas de la placa al otro extremo del tubo. Luego un rayo catódico barre la placa, cargándola, y las células fotoeléctricas emiten una carga eléctrica en función de la cantidad de luz recibida. La señal de imagen resultante se envía y amplifica para ser reproducida en un monitor.

Ipv4 (Internet Protocol Version 4): Cuarta versión del Internet Protocol (IP), uno de los principales protocolos de los métodos estándares de interconexión de redes basados en Internet. Utiliza direcciones de 32 bits, muchas de las cuales están dedicadas a redes locales en su especificación.

JavaScript: “Es un lenguaje de programación ligero e interpretado, orientado a objetos con funciones de primera clase que es más conocido como el lenguaje de script para

páginas web, aunque también se puede usar en entornos sin navegador. Es un lenguaje script multi-paradigma, basado en prototipos, dinámico, soporta programación funcional, orientada a objetos e imperativa” (MDN Web Docs Mozilla, 2018). Se utiliza principalmente del lado del cliente, el soporte está implementado como parte de un navegador web.

JSP (Java Server Pages): Tecnología que provee de una forma simplificada y rápida para crear contenido web dinámico. Permite un rápido desarrollo de aplicaciones basadas en la web que son independientes de la plataforma del servidor. Utiliza el lenguaje de programación Java y requiere un servidor compatible con contenedores servlet como Apache Tomcat.

Lipsync: Término de la industria de la televisión que se refiere a la sincronía entre la imagen y el audio. Deriva de los términos en inglés lips que es labios y sync de sincronía. Generalmente es más fácil detectar la sincronía de audio y video si el movimiento de los labios del interlocutor a cuadro corresponde al audio, cuando esto no ocurre se dice que la señal está fuera de lipsync.

Log: Archivo de bitácora de un sistema o aplicación, es un archivo que contiene las acciones más importantes o representativas así como operaciones críticas o relevantes tanto para llevar un registro de actividades del usuario o el software como con la finalidad de identificar anomalías en la programación o las acciones del usuario cuando hace algo indebido. Con base a las bitácoras se pueden detectar y corregir errores en el software.

Luminancia: Información relacionada con la mayor o menor luminosidad de la imagen y no contiene información alguna acerca de los colores de la misma. La luminancia por sí sola reproduce la imagen en blanco y negro con todas las tonalidades de gris intermedias de una imagen.

MOV: Culturación (2012) indica que es el formato para encapsular audio y video desarrollado por Apple originalmente para el reproductor QuickTime. Surgió en 1991 y ha evolucionado hasta convertirse en una de los formatos más utilizados en el mundo con soporte en editores y reproductores de video profesionales.

MPEG: Grupo de trabajo formado por la Organización Internacional de Normalización (ISO) y la Comisión Electrotécnica Internacional (IEC) con el objetivo de desarrollar estándares para la representación codificada de audio y video digital así como datos relacionados con la transmisión. Desde 1988 cuando se fundó el grupo de trabajo ha producido estándares que ayudan a la industria a ofrecer a los usuarios una mejor y más disfrutable experiencia digital. El grupo incluye compañías de la industria, universidades e instituciones de investigación.

NTSC (National Television System Committe): Sistema de televisión analógico empleado en Norteamérica, América Central y parte de América del Sur así como Japón. El comité fue establecido en julio de 1940 para proponer los estándares del sistema de televisión analógica en Estados Unidos.

Pauta: Concepto utilizado en la industria de la televisión para referirse a los comerciales que están planeados transmitir para cumplir con los compromisos adquiridos con los clientes o anunciantes y las regulaciones gubernamentales.

PCM (Pulse Code Modulation) o modulación por impulsos codificados: Procedimiento de modulación utilizado para transformar una señal analógica en una secuencia de bits o señal digital, que fue inventado por el ingeniero británico Alec Reeves en 1937 y es la forma estándar de audio digital en computadoras, telefonía digital y aplicaciones similares. El método consiste en que la amplitud de una señal analógica es muestreada regularmente en intervalos uniformes, cada muestra es cuantizada al valor más cercano de un rango de pasos digitales. Tiene dos propiedades básicas la frecuencia de muestreo y la profundidad de bits.

PHP: “(Acrónimo recursivo de PHP: Hypertext PreProcessor) es un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web y puede ser incrustado en HTML” (PHP Group, 2001).

Pillar box: Elemento necesario incluir en la visualización de un material de video que no ha sido diseñado para ser reproducido en una pantalla ancha o panorámica se requiere mostrar en un dispositivo con estas características. El espacio que no es capaz de llenar el contenido en el formato original preservando la relación de aspecto se rellena utilizando unas barras laterales, a estos elementos se les llama pillar box. Algunos ingenieros llaman pillar box al proceso de ajuste para el despliegue descrito.

Pipe de datos: Un pipe de datos en el sistema operativo Windows es una sección de memoria compartida que usan los procesos para comunicarse. El proceso que crea el pipe es llamado servidor del pipe. El proceso que se conecta a un pipe es un cliente del pipe. En este mecanismo de comunicación interproceso un proceso escribe información en el pipe y posteriormente el otro proceso lee la información desde el pipe.

Pleca: En la industria de la televisión es un elemento de diseño que se sobrepone a una porción de la pantalla y puede tener transparencia. Estrictamente desde el punto de vista del diseño una pleca es una barra o banda a lo ancho de la pantalla que incluye información.

Pre-roll: En la industria de la televisión esta palabra se refiere a la preparación de un material para ser transmitido. Originalmente los materiales estaban en cinta magnética, por lo que tenían que ser insertados los carretes de algún dispositivo de reproducción lo cual constituía la fase de preparación del material, lo cual originó el uso del término.

Producción: En la industria de la televisión una producción es un programa de televisión.

Promocional: Material de video que se transmite como parte de la señal de una televisora, principalmente en cortes comerciales, contiene información acerca de programación del canal de televisión, las diferentes producciones televisivas, avisos importantes o cualquier elemento que promueva entre la audiencia información que haga atractiva la señal del canal en el cual está incluido el promocional.

QuickTime: Framework multimedia desarrollado por Apple que consiste en un conjunto de bibliotecas y un reproductor multimedia.

Rango dinámico: En el audio, es la diferencia de nivel de presión sonora entre los sonidos más bajos y los sonidos más altos que se escuchan en una señal. Generalmente la diferencia se mide en dB y cuanto mayor sea mayor será el rango dinámico. Las ventajas de sonido con un rango dinámico más alto desde el punto de vista de la producción musical son un sonido más limpio y sin saturación, con muchos matices que permiten distinguir claramente instrumentos y voces, así como un mayor realismo del audio.

RGBA (acrónimo de Red, Green, Blue, Alpha): Espacio de color, o también se define como el modelo de color RGB más un canal alpha que almacena información de transparencia. Cada componente generalmente se almacena en 8 bits y cada pixel en 4 bytes. El canal alpha se usa como la opacidad del pixel, si contienen un valor de 0 es totalmente transparente y si tiene el máximo valor posible es totalmente opaco. Este espacio de color permite la composición de imágenes.

SDI (Serial Digital Interface) o Interfaz Digital Serial: Norma de interfaces para video digital que se estandarizó inicialmente por la SMPTE (Society of Motion Picture and Television Engineers) en 1989. Su principal aplicación es la transmisión de señal de video sin compresión o en banda base. Es un estándar disponible únicamente para dispositivos profesionales y entornos de producción que requieren operaciones a distancias cortas, 30 metros cuando máximo, y tasas de bits muy elevadas manejando datos sin compresión.

SDK (Software Development Kit): Conjunto de herramientas para el desarrollo de software que permiten al programador crear aplicaciones.

SMTP (Simple Mail Transfer Protocol) o Protocolo Simple de Transferencia de Correo: Protocolo TCP/IP que se utiliza para enviar y recibir correos electrónicos.

Socket: Concepto abstracto mediante el cual dos programas pueden intercambiar flujos de datos. Constituye un mecanismo para la entrega de datos provenientes de una tarjeta de red a los procesos adecuados. Un socket está definido por un par de direcciones IP, local y remota, un protocolo de transporte y los puertos tanto local como remoto.

Spot: En el ámbito de la televisión es cualquier material audiovisual que puede ser incluido en la transmisión dentro de un bloque comercial cuyo contenido puede ser branding del canal, promocional, información gubernamental o el comercial de un anunciante que ha contratado un espacio publicitario.

Súper: En la industria de la televisión es un contenido que se sobrepone en la imagen de la señal ocupando una porción de la pantalla y puede tener transparencia. La transmisión de un súper tiene la finalidad de presentar parte del branding del canal, avisos, información importante, promocionales o publicidad contratada por los anunciantes. Gran parte de su uso se da en transmisiones en vivo.

TGA: Acrónimo de Truevision Advanced Raster Graphics Adapter es un formato de archivo para guardar gráficos diseñado por Truevision, empresa que fabricaba una de las primeras familias de tarjetas gráficas para computadoras personales. El formato es compatible con profundidades de 8, 16, 24 o 32 bits por píxel con un máximo de 24 bits para colores RGB y un canal alfa de 8 bits. Tiene soporte para utilizar compresión sin pérdidas.

Thread o hilo de ejecución: Subproceso que se encarga de realizar tareas específicas, esta técnica de programación permite a una aplicación llevar a cabo varias funciones simultáneamente. Los threads comparten recursos como el espacio en memoria y archivos. Cuando el proceso principal termina también lo hacen todos sus subprocesos.

Timecode: Conjunto de estándares que sirven para etiquetar cada uno de los cuadros de un video e identificarlo usando una estampa de tiempo que incluye horas, minutos, segundos y cuadros en un formato donde cada cantidad tiene dos dígitos separados por “.”. Fue inventado a finales de la década de 1960 de forma que los sistemas de cómputo para la edición de video pudieran encontrar automáticamente cuadros específicos en la cinta de video y guardar las acciones del editor de manera que se pudieran repetir las secuencias. La SMPTE (Society of Motion Picture and Television Engineers) estandarizó muchos de formatos de timecode en 1967.

Transcodificación: Procesamiento que se aplica a un archivo digital de video para convertir su contenido de un formato a otro con la finalidad de permitir que el video se pueda reproducir en diferentes plataformas o dispositivos. Generalmente el proceso incluye dos pasos, en la primera parte el contenido se decodifica de manera que los datos originales son extraídos a un formato sin compresión. En la segunda parte los datos sin compresión se codifican en el formato deseado.

Try-Sync: Señal de video con pulsos de sincronización de tres niveles. Se usa en escenarios con señales en formato HD en vez de la Black Burst que se prefiere en SD, esto debido a que las variaciones de tiempo son muy reducidas dada su naturaleza de alta frecuencia. Por otro lado no contiene una componente de directa, los pulsos abarcan ambas polaridades.

Tubo rayos catódico: Dispositivo que permite visualizar imágenes mediante un haz de rayos catódicos emitido por un cañón de electrones que se dirige contra una pantalla de vidrio cubierta de fósforo y plomo. El fósforo permite reproducir la imagen proveniente del haz de rayos catódicos y el plomo bloquea los rayos X para proteger al usuario de las radiaciones. La capa de fósforo se ilumina cuando los electrones chocan con ella formando los píxeles de la imagen. En los últimos modelos se tenían 3 cañones de electrones y la pared con la que chocaban estaba dividida en pequeños círculos o líneas que contienen cada uno de los colores del espacio RGB. Los cañones realizan barridos de toda la pantalla a cierta velocidad con la cual dan la impresión al ojo humano de construir una imagen fija durante cierto tiempo. Los haces de electrones eran dirigidos por medio de bobinas electromagnéticas situadas en el interior del tubo.

UDP (User Datagram Protocol): Protocolo de nivel de transporte basado en el intercambio de datagramas, permite el envío de datagramas o paquetes de datos a través de la red sin que se haya establecido previamente una conexión, esto se puede hacer porque el propio datagrama incorpora suficiente información en la cabecera acerca del destino del paquete. Tampoco incluye confirmaciones de la recepción de los paquetes o control de flujo, por lo que algunos paquetes se pueden adelantar a otros. Es de amplio uso en la transmisión de audio y video en tiempo real, porque en estos escenarios no es posible realizar retransmisiones de paquetes debido a los estrictos requisitos del retardo, para enfrentar los problemas de transmisión en estas aplicaciones se recurre a mecanismos de detección y corrección de errores en una capa superior. El protocolo fue creado en 1980 publicado como el estándar RFC 768.

Up-conversion: Proceso para convertir una imagen en resolución menor, generalmente en formato SD, a una resolución mayor, generalmente en formato HD. Para esto se pueden utilizar diferentes métodos como el LetterBox o aplicar pillar boxes.

Valor RMS o valor eficaz: Valor cuadrático medio de una magnitud eléctrica o señal, el concepto se utiliza especialmente para estudiar formas de onda periódicas. Al ser un valor medio como por definición se indica, el valor eficaz es una magnitud representativa del efecto que causa la onda si esta pudiera ser reemplazada por una señal constante.

VTR (Video Tape Recorder): Dispositivo conocido como videocasetera utilizada aún en aplicaciones profesionales diseñada para reproducir audio y video desde una cinta magnética. Actualmente se pueden encontrar este tipo de dispositivos con soporte para formatos de video en HD e incluso soporte para grabar y reproducir contenido en MPEG2.

Watchdog (Perro guardián): Mecanismo de seguridad que provoca un reset del sistema en caso de que éste se haya bloqueado. Consiste generalmente de un temporizador que decrementa un contador que se ha cargado previamente con un valor alto, cuando el contador llega a cero se reinicia el sistema o se aplican las acciones pertinentes para lo cual está diseñado. Por lo tanto se debe contar con un mecanismo que recargue o reinicie el perro guardián antes de que el contador llegue a 0 para garantizar el funcionamiento del sistema supervisado.

Referencias

Advanced Television Systems Committee. (7 de Agosto de 2009). *ATSC Digital Television Standard: Part 4 – MPEG-2 Video System Characteristics*. Recuperado el 18 de Septiembre de 2018, de https://www.atsc.org/wp-content/uploads/2015/03/a_53-Part-4-2009.pdf

Apple Inc. (13 de Octubre de 2016). *Performing Alpha Compositing Operations*. Recuperado el 8 de Septiembre de 2018, de <https://developer.apple.com/library/archive/documentation/Performance/Conceptual/vImage/AlphaCompositingOperations/AlphaCompositingOperations.html>

Benito, F. (6 de Octubre de 2016). *¿Cuál es la diferencia entre 4K y Ultra HD?* Recuperado el 10 de Septiembre de 2018, de <https://omicro.no.espanol.com/2016/10/diferencia-entre-4k-y-ultra-hd/>

Blackmagic Design. (Diciembre de 2017). *Software Developers Kit - Decklink Versión 10.9.10*. Recuperado el 15 de Enero de 2018, de <https://www.blackmagicdesign.com/support/>

Burke, A. (2018). *¿Qué es el sonido PCM?* Recuperado el 3 de Octubre de 2018, de https://techlandia.com/sonido-pcm-sobre_128723/

Christensson, P. (2006). *Bitrate Definition*. Recuperado el 12 de Septiembre de 2018, de <https://techterms.com/definition/bitrate>

Culturacion. (2012). *¿Qué es un archivo MOV?* Recuperado el 9 de Septiembre de 2018, de <http://culturacion.com/que-es-un-archivo-mov/>

Davis, C. (2008). *Video Aspect Ratios --- Understanding Pillarboxing & Letterboxing*. Recuperado el 10 de Octubre de 2018, de <https://chetdavis.typepad.com/chet-davis/2009/11/pillarboxing-letterboxing-and-postage-stamp-effects-.html>

DVB. (Febrero de 2018). *Specification for the use of Video and Audio Coding in Broadcast and Broadband Applications*. Recuperado el 1 de Octubre de 2018, de https://www.etsi.org/deliver/etsi_ts/101100_101199/101154/02.04.01_60/ts_101154v020401p.pdf

Geekland. (13 de Enero de 2017). *¿Qué es el rango dinámico de un audio y que ventajas proporciona?* Recuperado el 21 de Septiembre de 2018, de <https://geekland.eu/que-es-el-rango-dinamico-audio-ventajas/>

Glosario de Internet e informática. (23 de Junio de 2017). *Glosario de Informática e Internet*. Recuperado el 10 de Septiembre de 2018, de <https://www.internetglosario.com/>

Iacobone, E. (12 de Julio de 2012). *Marcelo Coggan de Saeta TV: Sistema propio de automatización del master de emisión*. Recuperado el 27 de Noviembre de 2018, de <https://www.produ.com/television/noticias/marcelo-coggan-de-saeta-tv-sistema-propio-de-automatizacion-del-master-de-emision>

ISO/IEC. (2000). *International Standard ISO/IEC 13818-1*. Recuperado el 10 de Octubre de 2018, de <http://ecee.colorado.edu/~ecen5653/ecen5653/papers/iso13818-1.pdf>

Latin Press. (11 de Julio de 2008). *Automatización de estaciones de televisión en América Latina*. Recuperado el 27 de Noviembre de 2018, de <https://www.tvyvideo.com/200807113724/articulos/tecnologia/automatizacie-estaciones-de-televisin-amca-latina.html>

Martin, L. M. (Junio de 2017). *Glosario de Abreviaturas del Módulo: Sistemas de Radio y TV*. Recuperado el 6 de Agosto de 2018, de <http://platea.pntic.mec.es/~lmarti2/glosari1.htm>

MDN Web Docs Mozilla. (27 de Noviembre de 2017). *HTML*. Recuperado el 10 de Octubre de 2018, de <https://developer.mozilla.org/es/docs/Web/HTML>

MDN Web Docs Mozilla. (Septiembre de 2018). *JavaScript*. Recuperado el 10 de Octubre de 2018, de <https://developer.mozilla.org/es/docs/Web/JavaScript>

Microsoft. (30 de Mayo de 2018). *Pipes*. Recuperado el 8 de Septiembre de 2018, de <https://docs.microsoft.com/en-us/windows/desktop/ipc/pipes>

Newsline Report. (30 de Octubre de 2018). *LA AUTOMATIZACIÓN DEL MASTER CONTROL RESULTA ESTRATÉGICA PARA LAS OPERACIONES BROADCAST*. Recuperado el 27 de Noviembre de 2018, de <http://www.newslinereport.com/tecnologia/nota/la-automatizacin-del-master-control-resulta-estrategica-para-las-operaciones-broadcast>

Oracle. (2018). *Database Links*. Recuperado el 7 de Agosto de 2018, de https://docs.oracle.com/cd/B28359_01/server.111/b28310/ds_concepts002.htm#ADMIN12092

Palazzesi, A. (20 de Mayo de 2010). *Disco de Nipkow: TV mecánica (1884)*. Recuperado el 6 de Septiembre de 2018, de <https://www.neoteo.com/disco-de-nipkow-tv-mecanica-1884/>

PHP Group. (2001). *¿Qué es PHP?* Recuperado el 20 de Octubre de 2018, de <http://php.net/manual/es/intro-what-is.php>

Reyes, L. A. (1 de Abril de 2016). *Valor promedio, valor máximo, valor pico a pico y valor eficaz*. Recuperado el 25 de Noviembre de 2018, de <https://ingenieriaelectronica.org/valor-promedio-valor-maximo-valor-pico-a-pico-y-valor-eficaz/>

Román, R. G. (10 de Febrero de 2016). *¿Qué es para ti el branding?* Recuperado el 8 de Octubre de 2018, de <http://www.branderstand.com/que-es-para-ti-el-branding/>

Sencore. (21 de Octubre de 2011). *So what exactly is “video transcoding”?* Recuperado el 9 de Septiembre de 2018, de http://www.sencore.com/company/blog/so-what-exactly-%E2%80%9Cvideo-transcoding%E2%80%9D#.W_pX4ocza00

Textos Científicos. (27 de Julio de 2005). *SEÑALES EN BANDA BASE*. Recuperado el 15 de Octubre de 2018, de <https://www.textoscientificos.com/redes/senales/banda-base>

The Moving Picture Experts Group. (Febrero de 2002). *MPEG-4 Overview - (V.21 – Jeju Version)* . Recuperado el 20 de Octubre de 2018, de <https://mpeg.chiariglione.org/standards/mpeg-4>

W3C. (2016). *HTML & CSS*. Recuperado el 10 de Agosto de 2018, de <https://www.w3.org/standards/webdesign/htmlcss>