



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

SISTEMA DE APOYO EN LA PLANEACIÓN DE LA
PROGRAMACIÓN EN TELEVISORAS MEXICANAS

T E S I S

QUE PARA OBTENER EL TÍTULO DE
INGENIERO EN COMPUTACIÓN

P R E S E N T A N :

JOSÉ MARTÍN PATRICIO OROZCO

MARLENE PÉREZ FERNÁNDEZ

DIRECTOR DE TESIS:

M.I AURELIO ADOLFO MILLÁN NÁJERA



México, D.F. a 18 de noviembre de 2015.

Dedico esta tesis a mi familia que me apoyó en el proceso de titulación, especialmente a mis padres (Natividad y Reyes) quienes me dieron su cariño y me proporcionaron las herramientas para estudiar y concluir mis estudios universitarios, lo que me permitirá competir en la vida. Son ejemplo de vida ya que sin contar con estudios lograron una vida cómoda y formaron una familia con sólidas bases de valores morales.

Marlene Pérez Fernández.

A mi madre María de los Ángeles Orozco y mi hermana Cristina Patricio Orozco, dos de mis valiosos pilares en mi vida.

A todos mis amigos que hacen de mi vida una maravillosa experiencia.

J.Martín Patricio O.

Índice

Introducción	1
--------------------	---

Capítulo I

ANTECEDENTES

1.1	Ingeniería de Software.	
1.1.1	Definición y características.....	5
1.1.2	Paradigma de la ingeniería del software.....	6
1.2	Definición y concepto de base de datos.	
1.2.1	Definición de base de datos.....	8
1.2.2	Sistemas de gestión de bases de datos.....	9
1.2.3	Estándares ODBC y OLEBD	10
1.3	Business Intelligence.	
1.3.1	Concepto de Business intelligence.....	13
1.3.2	Extracción, transformación y carga (ETL)	17
1.3.3	Análisis y toma de decisiones.....	21
1.4	Medición de las variables de televisión en México.	
1.4.1	Nielsen IBOPE.....	29
1.4.2	Conceptos.....	30

Capítulo II

DEFINICIÓN DEL SISTEMA

2.1	Definición del problema.....	33
2.2	Análisis de requerimientos de datos.....	34
2.3	Análisis de requerimientos funcionales.....	42

Capítulo III

DESARROLLO DE LA APLICACIÓN

3.1	Extracción de datos	49
3.2	Transformación	60
3.3	Modelo asociativo.....	61
3.4	Desarrollo de la aplicación.....	70
3.5	Validación de datos y funcionalidad.....	80

Capítulo IV

IMPLANTACIÓN Y CIERRE

4.1	Configuración de ambiente productivo.....	83
4.2	Documentación técnica	91
4.3	Pruebas de usuario final	100
4.4	Capacitación	102
4.5	Documento de cierre.....	113
	Conclusiones.....	116
	Anexo A	118
	Anexo B	124
	Anexo C.....	134
	Bibliografía.....	138

Introducción

Las televisoras mexicanas cuentan con fuentes de información de diversas tecnologías e invierten mucho tiempo en la generación de reportes para que sus directivos puedan analizar su audiencia y después decidir qué programación transmitir.

Acceder a la información de las televisoras es una tarea difícil por las siguientes razones:

- a) No existe homologación entre las fuentes.
- b) Las fuentes no se encuentran ubicadas en la misma dirección física.
- c) Sólo está disponible a través de un software licenciado.

Los recursos de hardware resultan deficientes para poder manipular y analizar grandes volúmenes de datos (especialmente históricos).

La relación entre las áreas informáticas (constructoras de software) y las directivas (toma de decisiones) se ve afectada por la dependencia que existe entre ellas y por el tiempo de respuesta a determinadas peticiones.

Conscientes de todos estos inconvenientes que sufren las empresas televisoras se desarrollará el siguiente trabajo en el cual se propondrá un sistema que facilite el uso de la información generando reportes más rápidos, visuales y de fácil manejo para las empresas. Con este sistema, será más fácil lograr una visión global, incluso cuando se trabaja con enormes conjuntos de datos o en su caso datos muy complejos, permitiendo integrar información procedente de diversas fuentes y ponerla a disposición del usuario a través de la red y dispositivos móviles. Nos apoyaremos en la tecnología asociativa que permitirá crear una interfaz única para la presentación interactiva y el análisis de cualquier tipo de información.

Los sistemas convencionales de búsqueda de datos a menudo requieren una estructura fija, que va de lo más amplio a lo más detallado en las consultas. La recuperación de datos suele ser una tarea bastante compleja, que requiere un amplio conocimiento de la estructura de la base de datos, así como de la sintaxis del lenguaje de consulta. Muchas veces, el usuario se ve limitado por las rutinas de búsqueda predefinidas.

El sistema a utilizar, permitirá comenzar por cualquier parte de la información, sin importar su ubicación en la estructura de datos. Hay varios métodos, sin embargo Qliktech que es la propietaria del software que utilizaremos (Qlikview) ofrece una alternativa rápida

, confiable y de bajo costo para el manejo de la información de los negocios, la cual se basa en la metodología SAFE (Simplifying Analysis For Everyone) que hace posible un análisis simplificado de las mediciones de interés de cada empresa, en este caso televisoras mexicanas y para todos los involucrados en el proyecto. El software revoluciona este proceso, haciendo posible la libre selección de los datos para poder mostrarlos en una pantalla diseñada al gusto del usuario y con una interfaz fácil de manejar.

Como se sabe los sistemas de negocios permiten:

- ❖ Crear una interfaz flexible para los usuarios finales y un almacén de datos.
- ❖ Obtener rápidamente un retrato real de las relaciones entre los datos.
- ❖ Elaborar presentaciones basadas en sus datos.
- ❖ Crear objetos dinámicos e ilustrativos como tablas y gráficos.
- ❖ Analizar estadísticas.
- ❖ Vincular descripciones y funcionalidad multimedia a sus datos.
- ❖ Crear tablas nuevas, integrando información de fuentes muy diversas.
- ❖ Construir su propia estructura de inteligencia de negocio.

Algunos ejemplos donde se utilizan hoy en día son: sistemas financieros, administraciones de recursos humanos, análisis de mercados, atención al cliente, administración de proyectos, controles de producción, compañías de almacenes y compraventas. Incluso se pueden mezclar las diferentes aplicaciones entre sí para lograr visiones globales de la información, totalmente novedosas.

En el capítulo 1 se establecerán los antecedentes necesarios para entender ¿Qué son los sistemas de inteligencia de negocios?, ¿Cómo la ingeniería del software nos permite crear dichos sistemas?, relacionándolos con las diferentes bases de datos que actualmente existen y los estándares que la programación SQL utiliza en dichas bases de datos. También se hablará de ¿Cómo se miden las variables de las televisoras mexicanas?, ¿Quién las mide? y ¿Cómo se pueden utilizar las mismas?. Sobre todo reconocer que los métodos y procedimientos de la ingeniería del software han sido adoptados con éxito en un amplio espectro de aplicaciones industriales.

En el capítulo 2 tomaremos en cuenta las bases de la ingeniería del software y las necesidades que tienen las empresas de televisión en México al momento de presentar una gran cantidad de información a sus directivos.

Se realizará el análisis de los requerimientos de datos y de funcionalidad del sistema, iniciando con la identificación de la información, para ello es muy importante saber ¿Qué

datos se necesitan?, ¿Dónde se pueden encontrar? y ¿Cómo conseguirlos?. Una vez que se dispone de los mismos, se deben seleccionar aquellos que sean útiles para los objetivos del sistema y se deben preparar poniéndolos en un formato adecuado.

Para la creación del sistema utilizaremos Qlikview que es una plataforma que aporta simplicidad, escalabilidad y rendimiento gracias a que en la actualidad existe hardware de 64 bits con un consumo de recursos mucho menor, procesadores multi-núcleo y una caída constante en el costo de la memoria. Esta herramienta ofrece respuestas a las necesidades de los usuarios que quieren tener una analítica intuitiva y de rápido despliegue.

Nuestro sistema permitirá consolidar información y explorar datos de forma interactiva, ofreciendo objetos como tablas, gráficos, cuadros de lista para seleccionar dimensiones de los datos, cuadros de estadísticas y muchos otros elementos visuales en la interfaz. Será posible crear tableros de control y analizar distintas fuentes de datos obteniendo una visión general de la televisora. No importa ¿Cuántas? y ¿Cuáles? sean las fuentes de información. Será un sistema con una verdadera simplicidad que aportará respuestas al instante para todo tipo de usuarios, no sólo para unos cuantos expertos, además de que despertará su curiosidad y creatividad.

La funcionalidad exige la comprensión del dominio de la aplicación, del conocimiento relevante y de los objetivos del usuario final para poder presentar la información de una manera entendible que permita la interacción entre el sistema y el usuario final u otros sistemas.

En el capítulo 3 desarrollaremos nuestro sistema, el cual será fácil de manejar por los usuarios finales y enriquecerá sus facultades de análisis y presentación de la información de audiencia.

Llevaremos a cabo el proceso de extracción de datos de las fuentes de información, que para nuestro caso, serán archivos en formato .xlsx (Microsoft Office Excel).

Esta etapa es la más cercana a los entornos relacionales de donde se obtienen los datos. El objetivo es acceder a los datos origen necesarios y dejarlos preparados para el sistema, por ello debemos pensar en el detalle que queremos mostrar, la cantidad de datos que se utilizarán (histórico) y la redundancia entre ellos.

Teniendo los datos seleccionados, los transformamos al formato de Qlikview, se aplican las reglas de negocio y se dejan los indicadores calculados para que en la parte de desarrollo

de la aplicación se generen los objetos gráficos que permitan el manejo de expresiones las cuales deben ser lo más sencillas posibles.

Transformados los datos podremos unir las tablas obtenidas mediante el campo con mayor detalle, suele ser la llave primaria y así llegamos al modelo de datos o modelo asociativo.

El tipo de modelo de datos obtenido, impactará en el tiempo de ejecución del código del sistema (script) o en el tiempo de realización de una consulta por parte del usuario.

Una vez construido el sistema se realizarán pruebas de funcionalidad y validación de cifras con la finalidad de garantizar la integridad de los datos.

En el capítulo 4 pasaremos de un ambiente de desarrollo de prueba a un ambiente de desarrollo de producción (en tiempo real); se prepara la infraestructura física y lógica para la implantación y operación del sistema.

A pesar de que durante los capítulos precedentes se genera información que es documentada, en este capítulo se trabajará la documentación que normalmente se entrega a un usuario final (manual de usuario, documento de especificaciones técnicas, etc.).

Así mismo se sugiere una capacitación para el personal que utilizará el sistema, dicha capacitación contempla la navegación dentro de la aplicación, así como las reglas de la lógica del negocio. Y finalmente se propone un documento de cierre que generalmente contiene la aprobación de usuarios después de validar que el sistema cumple con sus requerimientos.

Capítulo I ANTECEDENTES

1.1. Ingeniería de software

1.1.1. Definición y características

La ingeniería de software difiere de la programación tradicional en que se utilizan técnicas de ingeniería para especificar, diseñar, codificar, validar y mantener los productos dentro del tiempo y presupuesto establecidos para un proyecto, además esta ingeniería se preocupa por aspectos administrativos que quedan fuera del dominio normal de la programación.

El término “programador” se emplea para denominar a la persona preocupada, abocada a las tareas y detalles de la modificación de los algoritmos, empaado y estructuras de datos codificados en algún lenguaje de programación particular.

Los ingenieros de software están, capacitados para hacer frente a aspectos de análisis, diseño, verificación, prueba de programas, la documentación, el mantenimiento y la administración del proyecto.

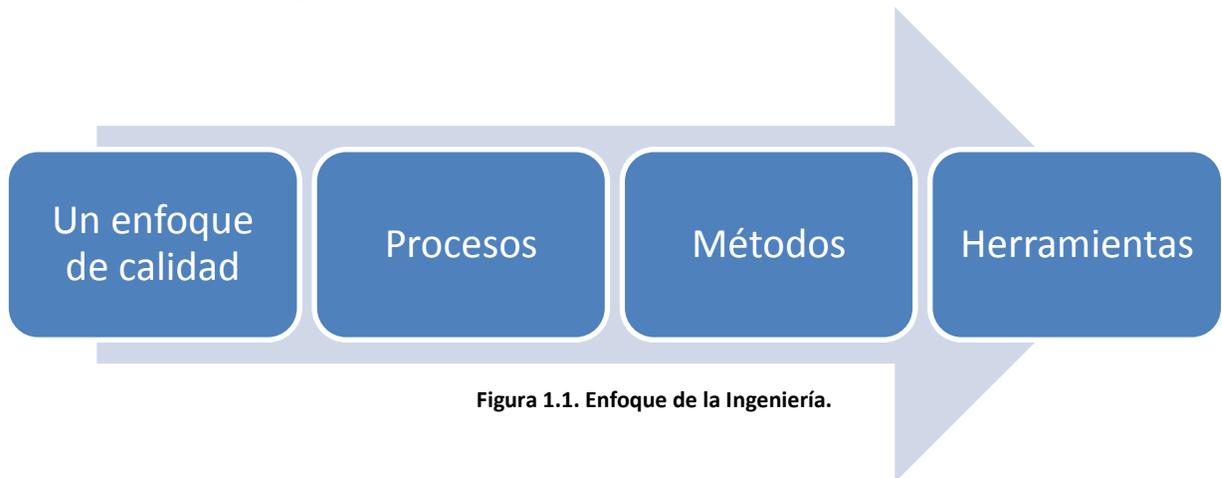


Figura 1.1. Enfoque de la Ingeniería.

La Ingeniería de software es una tecnología estratificada. Como se muestra en la figura 1.1, cualquier enfoque de la ingeniería debe estar sustentado en un compromiso con la calidad.¹

1. Figura 1.1. Tomada como base del contenido del libro. Roger S. Pressman. INGENIERÍA DEL SOFTWARE: UN ENFOQUE PRÁCTICO.

La base de la Ingeniería de Software es el proceso. El proceso es el elemento que mantiene juntos los estratos de la tecnología que permite el desarrollo racional y a tiempo del software de computadora. Define un marco de trabajo que debe establecerse para la entrega efectiva de la tecnología de la ingeniería del software además forma la base para el control de la gestión de los proyectos del software y establece el contexto en el cual se aplican los métodos técnicos, se generan los productos del trabajo (modelos, documentos, datos, reportes, etc), se establecen los fundamentos, se asegura la calidad y el cambio se maneja de manera apropiada.²

Los métodos abarcan un amplio espectro de tareas que incluyen la comunicación, el análisis de requisitos, el modelo del diseño, la construcción del programa y la realización de pruebas. Se basan en un conjunto de principios básicos que gobiernan cada área de la tecnología e incluyen actividades de modelado y otras técnicas descriptivas.

Las herramientas de la Ingeniería de Software proporcionan el soporte automatizado o semiautomatizado para el proceso y los métodos. Cuando las herramientas se integran de forma que la información que cree una de ellas puede usarla otra, se dice que se ha establecido un sistema para el soporte del desarrollo del software, que con frecuencia se denomina ingeniería del software asistida por computadora.

1.1.2. Paradigmas de la Ingeniería de Software

Un marco de trabajo establece la base para un proceso de software completo al identificar un número pequeño de actividades aplicables a todos los proyectos de software, sin importar su tamaño o complejidad. Además, abarca un conjunto de actividades aplicables a lo largo del proceso del software.

Consideramos que el siguiente marco de trabajo genérico del proceso se puede aplicar en la inmensa mayoría de los proyectos del software.

Comunicación: esta actividad del marco de trabajo implica una intensa colaboración y comunicación con los clientes;³ además, abarca la investigación de requisitos y otras actividades relacionadas.

2. Tomado del libro de Roger S. Pressman.

3. Un cliente es cualquier persona que tiene un interés en el éxito del resultado del proyecto: gerentes de negocios, usuarios finales, etc.

Planeación: esta actividad establece un plan para el trabajo de la ingeniería del software. Describe las tareas técnicas que deben de realizarse, los riesgos probables, los recursos que serán requeridos, los productos de trabajo que han de producirse y un programa de trabajo.

Modelado: esta actividad abarca la creación de modelos que permiten al desarrollador y al cliente entender mejor los requisitos del software y el diseño que logrará satisfacerlos.

Construcción: esta actividad combina la generación del código (ya sea manual o automatizado) y la realización de pruebas necesarias para descubrir errores en el código.

Despliegue: el software se entrega al cliente, quien evalúa el producto recibido y proporciona información basada en su evaluación.

Estas cinco actividades genéricas del marco del trabajo son útiles durante el desarrollo de programas sencillos, la creación de grandes aplicaciones en la red y en la ingeniería de sistemas basados en computadoras grandes y complejas.

La actividad de elaboración del modelado la componen dos acciones de la Ingeniería de Software: análisis y diseño. El análisis abarca un conjunto de tareas de trabajo (la investigación, la elaboración, negociación, especificación y validación de requisitos) que conducen a la creación del modelo de análisis. El diseño abarca tareas de trabajo (diseño de datos, diseño arquitectónico, diseño de interfaz y diseño al nivel de componentes) que crean un modelo de diseño.⁴

4. Cabe aclarar que “la elaboración del modelo” debe interpretarse de un modo diferente cuando se realiza el mantenimiento de un software existente. En algunos casos ocurre el modelado del diseño y el análisis, pero en otras situaciones de mantenimiento se utiliza para ayudar a entender el software heredado, al igual que para presentar adiciones o modificaciones en éste.

1.2 Definición y concepto de base de datos

Al hablar de la Ingeniería de Software se concluye que es una disciplina que integra el proceso, los métodos y las herramientas para el desarrollo del software de computadora. Pero este software debe estar encaminado a las bases de datos que es lo que se analizará y tomará en cuenta para poder implementar un sistema, por lo que es conveniente definir y hablar sobre las bases de datos. Empezaremos definiendo que es una base de datos, así como la importancia de considerar los sistemas de gestión de bases de datos. En la sección 1.2.3 se hablará del estándar ODBC desarrollado por Microsoft quien vino a estandarizar como se puede acceder a las bases de datos y del estándar OLEDB.

Importancia

La importancia de los sistemas de bases de datos se ha incrementado más con significativos desarrollos en lo que respecta a la capacidad del hardware, a la funcionalidad de éste y a las comunicaciones, incluyendo la aparición del internet, del comercio electrónico, de los sistemas de inteligencia empresarial. Los sistemas de bases de datos son, posiblemente, el desarrollo más importante en el campo de la Ingeniería de Software y las bases de datos forman ahora el marco de trabajo fundamental de los sistemas de formación, habiendo cambiado de forma significativa la manera en que muchas organizaciones operan.

Las bases de datos forman hoy en día una parte importante de nuestra vida cotidiana, hasta el punto que muchas veces no somos conscientes de estar usandola. Consideramos que una base de datos es una colección de datos relacionados y que el sistema de gestión de bases de datos (SGBD) es el software que gestiona y controla el acceso a los datos en algún punto de su ejecución.

Por lo que consideramos conveniente definir de una manera más formal las bases de datos y los sistemas de gestión de bases de datos.

1.2.1 Definición de base de datos

Base de datos { Es una colección compartida de datos lógicamente relacionados, junto con una descripción, que están diseñados para satisfacer las necesidades de información de una organización.

Un base de datos es un almacén centralizado, posiblemente de gran tamaño, compuesto por datos que pueden ser usados simultáneamente por múltiples departamentos y usuarios. En lugar de disponer de una serie de archivos desconectados con datos redundantes, todos los elementos de datos están integrados, manteniéndose al mínimo las

posibles duplicaciones. La base de datos deja de ser propiedad de un departamento y pasa a ser un recurso corporativo compartido, no sólo almacena los datos operacionales de la organización, sino también una descripción de dichos datos.

Los sistemas de bases de datos separan la estructura de los datos de los programas de aplicación y almacenan dicha estructura en la propia base de datos. Si se añaden nuevas estructuras de datos o se modifican las existentes, los programas de aplicación no se verán afectados, siempre y cuando no dependan directamente de la información que haya sido modificada. Por ejemplo, si añadimos un nuevo campo a un registro las aplicaciones existentes no se verán afectadas. Sin embargo, si eliminamos de un archivo un campo utilizado por un programa de aplicación, entonces dicho programa de aplicación sí se verá afectado por el cambio y deberá ser modificado correspondientemente.

Al analizar las necesidades de información de una organización, se trata de identificar entidades, atributos y relaciones. Una entidad es un objeto distintivo (una persona, lugar, cosa, concepto o suceso) dentro de la organización y que hay que representar en la base de datos. Un atributo es una propiedad que describe algún aspecto del objeto que queremos almacenar y una relación es una asociación entre entidades.

1.2.2 Sistemas de Gestión de Bases de Datos

SGBD { Sistema de software que permite a los usuarios definir, crear, mantener y controlar el acceso a la base de datos.

El SGBD es el software que interactúa con los programas de aplicación del usuario y con la base de datos. Normalmente proporciona la siguiente funcionalidad:

- Permite a los usuarios definir las bases de datos, usualmente mediante un lenguaje de definición de datos y especificar las estructuras, tipos de datos y las restricciones aplicables que hay que almacenar en la base de datos.
- Permite a los usuarios insertar, actualizar, borrar y extraer datos de las bases de datos, mediante un lenguaje de manipulación de datos. Al disponer de un depósito centralizado para todos los datos y descripciones de los mismos, este lenguaje puede proporcionar un mecanismo general de consulta de esos datos denominado lenguaje de consulta. El lenguaje de consulta más común es el lenguaje SQL (lenguaje estructurado de consulta), que es ahora el estándar formal como el estándar fijo para los SGBD.

- Proporciona un acceso controlado a la base de datos. Por ejemplo:
 - Un sistema de seguridad, que evita que los usuarios no autorizados accedan a la base de datos.
 - Un sistema de integridad, que mantiene la coherencia de los datos almacenados.
 - Un sistema de control de concurrencia que permite el acceso compartido a la base de datos.
 - Un sistema de control de recuperación, que restaura la base de datos a un estado previo coherente después de cada fallo de hardware o software.

1.2.3 Estándar ODBC (Conectividad abierta de base de datos)

El estandar ODBC es una interfaz de programación de aplicaciones (API) que forma parte de la programación en SQL. Es importante mencionar este estándar ya que forma parte y es uno de los recursos que hace uso del sistema a utilizar en nuestra investigación. La ODBC no es otra cosa que una técnica alternativa para incluir instrucciones SQL directamente en un lenguaje host, la cual consiste en proporcionar a los programadores una biblioteca de funciones que puedan ser invocadas desde el software de aplicación.

Para muchos programadores, la utilización de rutinas de biblioteca resulta bastante común, por lo que suelen considerar que el empleo de una API constituye una forma relativamente sencilla de utilizar SQL. Con esta técnica, en lugar de incluir instrucciones SQL en bruto dentro del código fuente del programa, el fabricante del SGBD proporciona una API que esta compuesta por un conjunto de funciones de biblioteca para muchos de los tipos más comunes de acceso a la base de datos que los programadores requieren, como por ejemplo, la conexión con la base de datos, la ejecución de instrucciones SQL, etc. Uno de los problemas de esta técnica, era la falta de interoperabilidad: era necesario procesar los programas utilizando el compilador proporcionado por el fabricante del SGBD y montar los programas con bibliotecas correspondientes a las API del fabricante.

Los fabricantes independientes de software se enfrentaban con un problema similar, al verse usualmente obligados a escribir una versión de cada aplicación para cada SGBD, o a escribir código específico de cada SGBD al que quisieran acceder. Esto implicaba que se tuviera que invertir una cantidad de recursos considerables desarrollando y manteniendo rutinas de acceso a los datos, en lugar de dedicar esos recursos a desarrollar y mantener las propias aplicaciones.

En un intento de unificar esta técnica, Microsoft especificó el estándar ODBC. Esta tecnología proporciona una interfaz común para acceder a bases de datos SQL heterogéneas, utilizando SQL como estándar de acceso a los datos. La interfaz (definida en el lenguaje C) proporciona un alto grado de interoperabilidad: una aplicación puede acceder a diferentes SGBD, SQL utilizando un mismo conjunto de código, lo que permite a los desarrolladores diseñar y distribuir aplicaciones cliente-servidor sin centrarse en un SGBD específico.

Una de las razones de la popularidad de ODBC es su flexibilidad:

- Las aplicaciones no están atadas a la API propietaria de ningún fabricante.
- Las instrucciones SQL pueden incluirse explícitamente en el código fuente o construirse dinámicamente en tiempo de ejecución.
- Las aplicaciones pueden prescindir de los detalles relativos a los protocolos de comunicación de datos subyacentes.
- Pueden enviarse y recibirse datos en un formato que resulta cómodo para la aplicación.

Estándar OLEDB (Vinculación e Incorporación de Bases de Datos)

OLEDB es una tecnología de base de datos de alto rendimiento basada en COM y creada por Microsoft. Lo que diferencia a OLEDB de otras tecnologías de base de datos de Microsoft es la forma en que proporciona acceso universal a los datos.

Las conexiones ODBC y OLEDB son las formas en las que una aplicación puede acceder a los datos en un banco de datos. Inicialmente las OLEDB se conocían como Vinculación e Incorporación de Bases de Datos (Object Linking and Embedding for Databases), pero ahora el acrónimo solo significa una interfaz de base COM para un amplio rango de orígenes de datos.

La estrategia de Microsoft de acceso universal a los datos, conocida como Universal Data Access, proporciona una forma común de obtener acceso a los datos sin importar la forma en que se almacenan. En una situación típica de negocios, una gran cantidad de información se almacena fuera de las bases de datos corporativas. Esta información se halla en sistemas de archivos (por ejemplo: FAT o NTFS), bases de datos personales (Microsoft Access), hojas de cálculo (Microsoft Excel), aplicaciones de planeación de proyectos (Microsoft Project) y correo electrónico (Microsoft Outlook).

El acceso a estos datos con las diferentes aplicaciones asociadas constituye un límite en la capacidad de transferencia de información de un sistema o una conexión, que puede reducir el tráfico en condiciones de sobrecarga. Suele producir una baja del rendimiento y la velocidad general tanto en un sistema como en una conexión. La mayoría de las compañías se encuentran en esta situación y se enfrentan al problema consolidando la información en un sistema de administración de bases de datos (DBMS).

La alternativa consiste en programar una solución Universal Data Access. Lo que implica dos características: la primera es una consulta distribuida o acceso uniforme a varios orígenes de datos (distribuidos) y la segunda es la posibilidad de poner los orígenes de datos en DBMS a disposición de las aplicaciones de base de datos.

Los orígenes de datos pueden ser o no del mismo. El acceso uniforme significa que se pueda ejecutar la misma consulta en todos los orígenes de datos con iguales resultados.

OLE DB permite desarrollar aplicaciones que obtienen acceso a diversos orígenes de datos, tanto si son DBMS como si no lo son. Hace posible el acceso universal a datos mediante interfaces COM compatibles con la funcionalidad DBMS de un origen de datos determinado. COM reduce la duplicación innecesaria de servicios y proporciona interoperabilidad máxima, no sólo entre orígenes de datos, sino también entre otras aplicaciones.

Dadas las controversias políticas de las decisiones de mercado de Microsoft, es importante conocer los pros y contras técnicos:

- Si el usuario quiere acceder a datos independientes del lenguaje SQL, debe migrar a OLE DB. Como se señaló, los conectores ODBC están ligados al lenguaje SQL. Si el problema del usuario se puede solucionar directamente manipulando las tablas, es mejor usar OLE DB.
- Si el usuario desea interfaces paralelas para datos multidimensionales, se recomienda OLE DB pues es el único estándar para datos y modelos con estas características, además reduce la cantidad de código requerido y tiempo de ejecución.
- Si el usuario requiere acceder a diferentes bases de datos (locales, compartidas, IOM, etc.) desde una sola aplicación, se debe usar OLE DB, de lo contrario se estarán usando rutas de código separada para cada método de acceso. Esto también es fundamentado en la idea de que los conectores OLE DB reducen las diferencias entre controladores, dialectos SQL y DBMS que han presentado problemas entre las configuraciones ODBC.

- Si la necesidad del usuario es ejecutar actualizaciones concurrentes, la configuración OLE DB es la mejor opción. Para actualizar registros usando un controlador ODBC, el usuario debe generar sentencias SQL UPDATE que son básicamente actualizaciones orientadas a lotes. No existe un concepto de bloquear registros en un modelo ODBC. La configuración OLE DB se acomoda a varios modelos permitiendo la administración de muchos más datos desde aplicaciones.

Es por todo lo anterior que para desarrollo de modelos Qlikview complejos y que dependen de múltiples bases de datos, se recomienda usar conexiones OLE DB en vez de conexiones ODBC.

1.3 Business Intelligence

En la sección 1.3.1 se esboza el concepto de Business Intelligence. En la sección 1.3.2 se explica el proceso de la extracción, transformación y carga (ETL) que es necesario para acceder a los datos de las fuentes de información que se encuentran en el almacén de datos (datawarehouse) y en la sección 1.3.3 se explica el análisis y toma de decisiones.

1.3.1 Concepto de Business Intelligence

En tiempos de incertidumbre financiera, la toma de decisiones basada en información es crucial para sobrevivir en el mundo de los negocios. Empresas de todas las industrias dependen de la información para tomar sus decisiones. A menudo, la información que necesitan es difícil de obtener dado que no reside en sus sistemas departamentales o simplemente está fragmentada en diferentes aplicaciones. El resultado es una visión aislada del negocio que dificulta en extremo la toma de decisiones. Para que una empresa alcance un desempeño óptimo, la información debe fluir a través de sus fronteras funcionales e impactar con valor a toda la compañía. Es indispensable definir Business Intelligence.

El Data Warehouse Institute define Business Intelligence (BI) como:

“La combinación de tecnología, herramientas y procesos que nos permiten transformar nuestros datos almacenados en información, esta información en conocimiento y este conocimiento es dirigido a un plan o una estrategia comercial.

La inteligencia de negocios debe ser parte de la estrategia empresarial, ya que le permite optimizar la utilización de recursos, monitorear el cumplimiento de los objetivos de la

empresa y aumenta la capacidad de tomar buenas decisiones para así obtener mejores resultados.”⁵

Las personas encargadas de la toma de decisiones dentro de una organización requieren acceder a todos los datos de la organización, independientemente de la ubicación de éstos.

Para poder realizar un análisis exhaustivo de la organización, de sus requisitos y de las tendencias que están por debajo, se debe poder acceder no sólo a los valores actualmente almacenados en la base de datos, sino también a los valores históricos. Para facilitar este tipo de análisis, se creó el concepto de almacén de datos para contener datos extraídos de diversas fuentes, mantenidos por diferentes unidades operativas, junto con las transformaciones históricas y los correspondientes resúmenes. Los almacenes de datos basados en tecnología de base de datos ampliada proporcionan los mecanismos para gestionar todo este cúmulo de información. A lo largo de los últimos años han surgido dos tipos de herramientas de análisis: las herramientas de procesamiento analítico en línea (OLAP) y las herramientas de minería de datos. En la figura 1.2, se muestra la arquitectura de inteligencia de negocios.

La inteligencia de negocios permite ofrecer alta disponibilidad de datos, aglutinando información como: datos originados en las operaciones habituales en las organizaciones (transacciones), información generada en la experiencia del personal y que permanecía sin registrar en ningún medio accesible y compartible, datos de seguimiento de procesos sin informatizar e impacto de anteriores decisiones o similares enriqueciendo el punto de vista actual.

5. Roger S. Pressman. INGENIERÍA DEL SOFTWARE: UN ENFOQUE PRÁCTICO. Sexta edición. Editorial: MC Graw Hill.

Arquitectura de Inteligencia de Negocios.

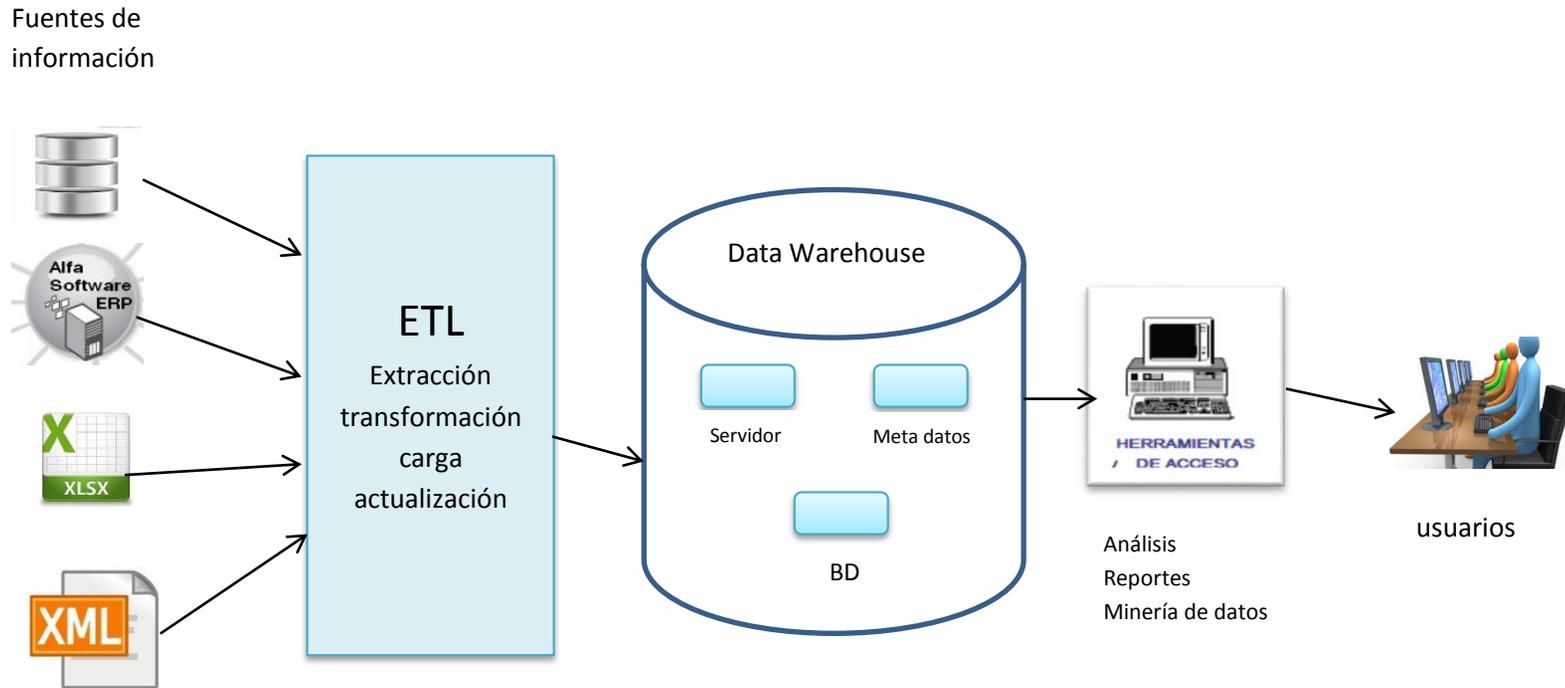


Figura 1.2 Arquitectura de Inteligencia de Negocios. ⁶

6. Imágenes tomadas de internet: taringa.net, <http://siskle.sisinfomanagement.com/articulo02.html>, ingsoftportafoliotematico.blogspot.com, glosarioinformatico3er.blogspot.com, informaticabi5.blogspot.com, recursostic.educacion.es.

Concepto de almacén de datos o Datawarehouse.

Almacén de datos { Es una colección de datos clasificada por temas, integrada, variable en el tiempo y no volátil que se utiliza como ayuda al proceso de toma de decisiones por parte de quienes dirigen una organización.

Según Inmon (1993), los datos:

- Están *clasificados por temas*, ya que el almacén de datos está organizado de acuerdo con los temas que más importancia tienen para la organización (clientes, productos y ventas) en lugar de organizarse por áreas de aplicación (facturación, control de almacén y pedidos). Esto se refleja en la necesidad de almacenar datos de ayuda en la toma de decisiones, en lugar de datos orientados a las aplicaciones.
- Están *integrados* debido a la mezcla de datos procedentes de diferentes sistemas de aplicación utilizados dentro de la organización. Los datos de origen son a menudo incoherentes, utilizando, por ejemplo, diferentes formatos. El almacén integrado de datos debe volver a dotarse de coherencia, para presentar una vista unificada de los datos a los usuarios.
- Son *variables en el tiempo* porque los datos del almacén de datos sólo son precisos y válidos en algún instante temporal o a lo largo de un cierto intervalo de tiempo.
- Son no volátiles, ya que los datos no se actualizan en el tiempo real sino que se refrescan en forma periódica a partir de los sistemas operacionales. Los nuevos datos se añaden siempre para aumentar la base de datos, en lugar de sustituir la información ya existente.

Procesamiento Analítico en Línea (OLAP)

En el apartado anterior se comentó acerca de los almacenes de datos, que combinan grandes volúmenes de éstos con el propósito de analizarlos. Sin embargo, junto al crecimiento del almacén de datos también se está incrementando la demanda de los usuarios para disponer de herramientas de acceso más potentes que proporcionen capacidades analíticas avanzadas. Hay dos tipos principales de estas herramientas: las de procesamiento analítico en línea (OLAP) y las de minería de datos. Estas dos herramientas constituyen la tecnología denominada inteligencia de negocios, de ahí la importancia de hablar de ellas.

En las últimas décadas, ha sido notable la creciente popularidad y prevalencia de los SGBD, de modo que una parte significativa de los datos corporativos se almacenan en dicho tipo de sistema. Las bases de datos se han estado utilizando principalmente para soportar sistemas de procesamiento de transacciones en línea (OLTP, Online Transaction Processing). Para proporcionar un soporte apropiado a los sistemas OLTP, los SGBD se han desarrollado de forma que permiten la ejecución eficiente de un gran número de transacciones relativamente simples.

Como se explicó anteriormente un almacén de datos conserva datos operacionales y puede soportar un amplio rango de consultas simples y complejas. Sin embargo, la capacidad de responder a consultas concretas depende de los tipos de herramientas avanzadas de acceso que los usuarios puedan utilizar en el almacén de datos.

Procesamiento analítico en línea (OLAP) { Es la síntesis, análisis y consolidación dinámicas de grandes volúmenes de datos multidimensionales.

Es un término que describe una tecnología que utiliza una vista multidimensional de datos agregados para proporcionar un rápido acceso a la información estratégica, con el propósito de realizar un análisis avanzado.

OLAP permite a los usuarios comprender mejor diversos aspectos de los datos corporativos, gracias a un acceso rápido, coherente e interactivo a una amplia variedad de posibles vistas de los datos. Mientras que los sistemas OLAP pueden también responder fácilmente a las preguntas del estilo de ¿Quién? Y ¿Qué?, en su capacidad de responder las preguntas de tipo ¿Qué pasaría si? y “¿Por qué?” lo que los distingue de las herramientas de consulta de propósito general.

Un cálculo típico puede ser mucho más complejo que una mera agregación de datos. Por tanto los tipos de análisis disponibles en las herramientas OLAP van desde las funciones básicas de navegación y exploración hasta la realización de cálculos y la realización de análisis más complejas, como series temporales y modelados de gran complejidad.

1.3.2 Proceso de extracción, transformación y carga (ETL)

En un almacén de datos toda fuente de información pasa por el proceso de extracción, transformación y carga, el proceso trata de recuperar los datos de las fuentes de información y así alimentar este almacén.

Esta parte del proceso de construcción del almacén de datos es costosa y consume una parte significativa de todo el proceso, por ello requiere recursos, estrategias, habilidades especializados y tecnologías.

El proceso de la extracción, transformación y carga es necesario para acceder a los datos de las fuentes de información al almacén de datos.

El proceso ETL se divide en 5 subprocesos⁷ que veremos a continuación:

1. Extracción

La extracción de los datos se puede realizar bien de forma manual o bien utilizando herramientas de ETL. De forma manual significa programar rutinas utilizando lenguajes de programación que extraigan los datos de las fuentes de datos origen, aunque en otros casos se opta por las utilidades de replicar la base de datos que tiene los motores de bases de datos. La alternativa más rentable es la que proveen las herramientas especializadas de ETL, ya que han sido diseñadas para llevar a cabo esta función, nos permiten visualizar y detectar los errores durante el proceso o durante la carga. Cada vez más los motores de bases de datos tienen mejores funcionalidades de ETL.

El principal objetivo de este subproceso es extraer tan sólo aquellos datos de los sistemas transaccionales que son necesarios y prepararlos para el resto de los subprocesos de ETL. Para ello se deben determinar las mejores fuentes de información, las de mejor calidad. Con tal finalidad, deberemos analizar las fuentes disponibles y escoger aquellas que sean mejores.

Normalmente hablamos de almacenes de datos intermedios (Data staging) mientras que estamos en el proceso de limpieza de los datos. Se trata de un paso intermedio entre la extracción y las etapas posteriores: acumulamos datos de distintas fuentes, en un momento determinado todos estos datos se cargarán en el datawarehouse. Los usuarios finales nunca acceden a este entorno.

2. Limpieza

Los sistemas transaccionales contienen datos que no han sido depurados y que deben ser limpiados. Las herramientas ETL tienen funcionalidades de limpieza de datos, aunque existen herramientas especializadas para ello. Si no llevamos a cabo este subproceso de

7. Roger S. Pressman. INGENIERÍA DEL SOFTWARE: UN ENFOQUE PRÁCTICO. Sexta edición. Editorial: MC Graw Hill.

forma exquisita, crearemos escépticos al mostrar los resultados si, por ejemplo, mostramos los mejores clientes de nuestra organización y aparecen duplicados; en tal caso, lo más habitual es que se cuestione la validez del modelo. Pero ¿cuáles son las causas que provocan que los datos estén “sucios”?, veamos algunos ejemplos:

- Captura errónea de la información.
- Uso inapropiado de los campos.
- Identificadores que no son únicos.
- Problemas de carga de antiguos sistemas o de integración entre sistemas.
- Selección del primer valor de una lista por defecto.

La limpieza de datos se divide en distintas etapas, que vamos a describir a continuación:

- **Depurar los valores:** este proceso localiza e identifica los elementos individuales de información en las fuentes de datos y los aísla en los ficheros destino. Por ejemplo: separa el nombre completo en nombre, primer apellido, segundo apellido, o separa la dirección en: calle, número, piso, etcétera.
- **Corregir:** este proceso corrige los valores individuales de los atributos usando algoritmos de corrección y fuentes de datos externas. Por ejemplo: comprueba una dirección y el código postal correspondiente.
- **Estandarizar:** este proceso aplica rutinas de conversión para transformar valores en formatos definidos aplicando procedimientos de estandarización y definidos por las reglas del negocio, sustituyendo los diminutivos de nombres por los nombres correspondientes.
- **Relacionar:** este proceso busca y relaciona los valores de los registros, corrigiéndolos y estandarizándolos, basándose en reglas de negocio para eliminar duplicados. Por ejemplo: identificando nombres y direcciones similares.
- **Consolidar:** este proceso analiza e identifica relaciones entre registros y los junta en una sola representación.

3. Transformación

La transformación de los datos se hace partiendo de los datos una vez “limpios”. Transformando los datos de acuerdo con las reglas de negocio y los estándares que han sido establecidos. La transformación incluye: cambios de formato, sustitución de códigos, valores derivados y agregados.

Los agregados normalmente se pre-calculan y se almacenan para conseguir mayores rendimientos cuando lanzamos las consultas que requieren el cálculo de totales al datawarehouse.

4. Integración

La última etapa es la de integración en el datawarehouse: es en esta etapa en la que se cargan los datos y se comprueban. Es fundamental comprobar que se ha desarrollado correctamente, ya que en caso contrario pueden llevar a decisiones erróneas a los usuarios.

5. Actualización

Este proceso determina la periodicidad con la que haremos nuevas cargas de datos al datawarehouse.

Herramientas ETL

En el proceso de extracción, transformación y carga, las herramientas ETL son claves en los proyectos de Business Intelligence. El mercado demanda herramientas ETL más completas y con más funcionalidades, que aceleren la extracción y carga de datos, que puedan acceder a diversos formatos y fuentes de datos, que soporten mayor complejidad y que se acerquen a cargas en tiempo real.

Las herramientas ETL deben contar con:

- **Diseño gráfico:** entorno que permite a los desarrolladores establecer la relación entre las fuentes de datos, las transformaciones, los procesos y las tareas para desarrollar la carga. Los diseños se deben almacenar en un depósito para metadata.
- **Gestión del metadata:** proveer un repositorio donde definir, documentar y gestionar la información del proceso ETL y su ejecución. El metadata debería ser accesible también desde otras aplicaciones.
- **Extracción:** extracción de la información mediante conectores, nativos de los distintos motores de bases de datos o ficheros planos. Los conectores deberían acceder al metadata para determinar ¿Qué información extraer? y ¿Cómo?.

- **Transformación:** deberían proveer de librerías de transformación que permitan a los desarrolladores transformar los datos origen en los destino con las nuevas estructuras y crear las tablas de agregación para mejorar el rendimiento.
- **Carga:** utilizar adaptadores para poder insertar o modificar los datos en el datawarehouse.
- **Servicios de transporte:** las herramientas ETL utilizan las redes y sus protocolos (por ejemplo: FTP, File Transport Protocol) para mover los datos entre las distintas fuentes y los sistemas destino.
- **Administración y operación:** las herramientas ETL deben permitir a los administradores programar, ejecutar y monitorear los trabajos de ETL, los resultados, gestionar los errores, recuperar los fallos y reconciliar los resultados con los sistemas originales.

1.3.3 Análisis y toma de decisiones

La toma de decisiones es un proceso que comienza reconociendo la existencia de un problema a tratar (por necesidad o imposición) culmina en una elección consciente y racional de una posibilidad entre diversas alternativas, a fin de resolver el problema. Cada opción debe ser debidamente evaluada, eligiendo una de ellas en función a las consecuencias previsibles por su implantación y su factibilidad de ocurrencia. Para una evaluación integral, el analista debe contar con información suficiente para analizar restricciones y objetivos condicionantes de cada opción: así podrá elegir la mejor de las alternativas.

La toma de decisiones se desarrolla mediante un proceso complejo en cuyo estudio convergen diferentes corrientes de pensamiento. En las comunidades tanto de negocios como técnica, las disciplinas de inteligencia de negocios y toma de decisiones aparecen como disociadas: los estudiosos del proceso de toma de decisiones resaltan y reclaman la necesidad de contar con datos variados, precisos y en tiempo, mientras que los técnicos argumentan que esta disciplina “es el respaldo a la toma de decisiones”.

No obstante haberse desarrollado bajo una realidad interdependiente, hace falta un ámbito de integración e interrelación práctico entre ambas técnicas, tanto en la industria, en los ámbitos de capacitación de las disciplinas administrativas como en las informáticas.

Su integración eleva el punto de mira del analista, pues aprovecha la beneficiosa concurrencia de ambas corrientes solidificando sus respectivos aportes conceptuales y

prácticos, un conocimiento de consolidado crecimiento y una estructura metodológica para el proceso de toma de decisiones.

Los elementos que constituyen la estructura de una decisión son:

- Objetivos considerados por quién decide.
- Restricciones para concretarlos, explicados mediante información abundante y precisa.
- Alternativas posibles y potenciales debidamente cuantificadas.
- Consecuencias de cada alternativa simulando diferentes escenarios.
- El escenario en el que se toma la decisión y preferencias de quién decide.

El crecimiento en contenido e importancia de la inteligencia de negocios son un fuerte aporte al mecanismo de toma de decisiones. La inclusión de una variedad de datos indujo a decisiones más seguras y permitió hacer un seguimiento más fino del comportamiento de las empresas, controlando desvíos en relación a objetivos estratégicos fijados oportunamente.

Cada decisión tiene factores circunstanciales condicionantes. Si la decisión resultó en un fracaso, se recomienda “no volver para atrás” sino “mirar para atrás” analizando las circunstancias que fallaron o no acompañaron las acciones consecuentes a la decisión. La BI permite analizar el resultado de una decisión.

La disciplina de la inteligencia de negocios construyó una magnífica estructura de consulta e investigación. Gracias a poderosos motores de consulta, el analista es liberado de cualquier restricción para el manejo de la información, gana una total independencia con respecto al centro de cómputo y puede plasmar los análisis tal como él lo hace en su realidad, con una mínima capacitación no técnica. Adicionalmente, las interfaces son tan polifacéticas y sencillas de usar que se revierte un camino existente: la información “va” hacia el analista y no al revés, como sucedía antes de su existencia.

Soluciones de la inteligencia de negocios ante exigencias de las decisiones. ⁸

1. Análisis transversal.

Satisface la necesidad propia del analista: “pensar los negocios”. No hace cortes segmentados de la empresa, sino que la ve como un todo. Puede relacionar datos registra-

8. Thomas M. Connolly, Carolyn E. Begg. SISTEMAS DE BASES DE DATOS: Un enfoque práctico para diseño, implementación y gestión. Cuarta edición. Editorial: PEARSON Addison Wesley.

dos en bases de datos diferentes (incluso tecnológicamente incompatibles). Para ello, la inteligencia de negocios permite satisfacer la necesidad de cruzar información no solamente interna, departamental sino externa, en formatos diferentes o proyecciones al futuro como presupuestos o tableros de comando.

2. Portabilidad.

Es sabido que el significado de los datos varía con extrema velocidad, lo cual es también cierto para la estructura de datos. Lo que ayer era importante, hoy es relativo. Lo que hoy es significativo mañana será reemplazado por otra estructura ¿Cómo puede la inteligencia de negocios adecuarse a dicha realidad?. Si una organización cambia de motor de alguna de sus bases de datos, un datawarehouse independiente puede seguir en funcionamiento. Esto asegura una alta disponibilidad de datos ante cambios de tecnología y su vigencia aunque el software originario se reemplace por otro más moderno. Esta posibilidad aporta un beneficio implícito adicional muy potente.

3. Democratización de la información.

La inteligencia de negocios permitió la democratización definitiva de la información, permitiendo que los miembros de un equipo compartan una misma visión, aún en zonas geográficas alejadas. Esta entidad conceptual cambió radicalmente la forma de trabajo en equipo: se acotó el entorno de lo discutible y se robusteció el proceso de la toma de decisiones. Contar con la misma base de información indujo a discutir solamente alrededor de estrategias o propuestas específicas. Y en muchos casos, esta puesta en común de información permitió detectar problemas semánticos en un mismo sector: distintos integrantes daban un significado diferente a una agrupación determinada de información.

Acceder a una fuente única facilita la discusión debido a que las refutaciones a ideas se hacen sobre los mismos gráficos de datos o vistas comunes a todos. De esta manera la inteligencia de negocios sirve como un medio de constatar y evaluar propuestas alternativas.

4. Recolección de datos y su capacidad formativa.

Anteriormente, el duro trabajo de la persona que decide incluía buscar la información, seleccionarla, depurarla, organizarla y transformarla para una fácil lectura y comprensión; hoy en día cualquier herramienta de inteligencia de negocios automatiza esta tarea con procesos de integración, depuración y transformación permitiendo acelerar los tiempos de dichos procesos en beneficio del proceso de decisiones, siempre exigido por la realidad.

Esto posibilitó que éstas pudiesen elaborarse con mayor calidad, con mayor tiempo para su elaboración, eliminando errores y minimizando el stress por falta de tiempo.

5. Capacidades analíticas independientes.

Tradicionalmente, los analistas solicitaban al centro de cómputo listados analíticos o estadísticos para completar la información estándar disponible. La dependencia era fuerte y no siempre se podía obtener una respuesta rápida. Los nuevos reportes eran “complementarios” a los disponibles y no se “integraban” a éstos.

Las herramientas de inteligencia de negocios solucionan este problema originario proveyendo a los analistas de un alto grado de independencia en su labor investigativa. Es fundamental su participación con el analista de sistemas al definir una estructura de datos que le permita aplicar cualquiera de sus análisis tradicionales o los que pueda crear: es quien mejor conoce las reglas del negocio y quien más experiencia tiene en la elaboración de decisiones.

Se refuerzan tres conceptos de independencia:

- Independencia de terceros.
- No requerir conocimientos técnicos informáticos.
- Libertad para poder ensayar cualquier tipo de análisis.

6. Interfaces.

Es fuerte la inversión de las empresas desarrolladoras de inteligencia de negocios en hacer que el producto sea rápidamente aceptado por gerentes de área, con nulo conocimiento en computación y con muy poco tiempo para aprender.

Justamente, estos sistemas están dirigidos a un usuario de particulares características:

- Poco tiempo para tomar decisiones.
- Poco tiempo para juntar información.
- Mucha presión para tomar decisiones.
- Poca disposición y tiempo para aprender técnicas de computación.
- Poca disposición y conocimientos para aprender técnicas de bases de datos.
- Poca disposición para incorporar nuevas herramientas.

Las herramientas de inteligencia de negocios superan estos inconvenientes presentando interfaces generosas en facilidades de uso: sencillez e intuición. Toda operación se hace con uno o dos clic del mouse, con variedad de formas para estructurar un mismo conjunto de datos. Esto permite un verdadero análisis, pues el usuario podrá “ver” un mismo conjunto de datos a través de diferentes puntos de vista o perfiles de análisis, con alto grado de integración o de detalle.

Esta facilidad brinda un entorno de trabajo diferente: el analista estaba acostumbrado a recibir la información en términos que él ya incorporó como hábito. En cambio, la combinación de datos con diferentes niveles de apertura transforma al producto en una herramienta de trabajo activa. Este grado de libertad fue siempre esperado para evitar interacciones con el personal informático especializado y realizable en cualquier momento.

7. Informes sistematizados.

El gerente puede estructurar un informe y conservarlo. Cada vez que se incorporan datos nuevos, se hacen visibles inmediatamente permitiendo que la calidad y productividad del gerente se potencien. Estos informes pueden ser enviados periódica y automáticamente a una lista predeterminada de personas, transformándose así en una herramienta activa.

8. Herramienta proactiva.

Con la inteligencia de negocios cambió la modalidad del uso de datos. En lugar de información en tiempo real, los encargados de tomar las decisiones (áreas sensibles a los cambios), requieren información que prevea el futuro. Respondiendo a esta necesidad la Business Intelligence ha variado su carácter de reactivo a proactivo. También es posible definir reglas de negocio que determinan el comportamiento estándar de los datos. La inteligencia de negocios verifica estas reglas y emite señales de su incumplimiento. Lo hace mediante semáforos administrados por el usuario o por medio del envío de email a los interesados en manejar el estado de excepción. Este comportamiento activo permite anticiparse a desvíos y poder tratarlos tempranamente.

9. Inteligencia competitiva.

La inteligencia competitiva trata de recabar, entender y categorizar las sensaciones y sentimientos de los clientes (reales o potenciales) sobre una compañía y sus productos, como también sobre sus competidores (y sus productos).

Desde la generalización del uso de recursos de la web 2.0 (chat, blogs, wikis, comunidades en línea) se puede acceder a un material inédito. En estos sitios, se habla de todo,

incluyendo sobre la organización y sus empresas competidoras, sus productos y sus servicios. Esto es una nueva oportunidad para que las organizaciones supervisen y lean dichos comentarios recolectando la sensación verdadera de los clientes.

Procesar manualmente toda esa información es obviamente casi imposible y su clasificación y presentación son tareas monumentales.

Los productos, procesos y servicios comprendidos dentro de la inteligencia competitiva lo hacen mucho más fácil como: sitios diseñados específicamente, salas de chat, las comunidades utilizan la taxonomía y la antología específica para identificar opiniones de consumidores.

Finalizada la búsqueda, el componente de inteligencia competitiva dentro de la Business Intelligence categoriza, tabula los resultados de la búsqueda y los presenta dentro de su datawarehouse. Dado que en estos centros generadores de información creciente, aparecen fuentes de información impensables. La futura explotación de esta información es invaluable y marca un nuevo camino a desarrollar.

10. Minimizar la incertidumbre.

Todo problema de negocio debe ser planteado como una pregunta para ser respondida utilizando la información disponible. La claridad y precisión en la definición del problema es fundamental. Toda decisión asume uno o varios riesgos, definidos como la posibilidad incierta de sufrir una pérdida.

Esta condición de incertidumbre es también señalada por Richard Daft cuando afirma que *“las decisiones pueden ser arriesgadas e inciertas, sin garantía alguna de éxito, pues deben hacerse en medio de factores en constante cambio, información poco clara y puntos de vista en conflicto”*. La incertidumbre crece cuando el número de alternativas es grande y todas tienen la misma oportunidad de ser elegidas.

La inteligencia de negocios está comprometida con el principio de la utilidad, por lo que para que cualquier decisión sea efectiva debe minimizar sus riesgos de error. No es siempre fácil identificar las variables relevantes que intervienen. La calidad de las definiciones está determinada por la calidad de las alternativas que se elijan. El entorno a una decisión es de una gran movilidad, por lo que su sostenimiento en el tiempo lo hace frágil requiriendo planificarse sobre la marcha. La Business Intelligence genera un flujo permanente de información confiable, organizada y presentada de tal manera que contribuye a reducir la

incertidumbre y la ambigüedad que rodea todo proceso de toma de decisiones, facilitando todas las acciones de planeamiento estratégico comercial anticipatorio.

11. Análisis de sensibilidad.

Se define análisis de sensibilidad como la técnica que examina el impacto causado a un sistema (conjunto de datos, programa informático, balance, etc.) al cambiar algún supuesto subyacente. La inteligencia de negocios ofrece posibilidades de aplicarlo: efectuar cambios en alguna de las variables y analizar la respuesta del modelo de información. Este verdadero análisis de comportamiento puede facilitar la selección de una premisa considerando todos los restantes aspectos del probable impacto de su selección, calculando el grado de incertidumbre y su moderación.

12. Inteligencia y anticipación.

Cuando se habla de “inteligencia” en Business Intelligence, se refiere a la capacidad requerida para transformar información obtenida en conocimiento nuevo, y ponerse frente a la posibilidad de generar soluciones útiles a una problemática dada. Muchas empresas se enfrentan cotidianamente al problema de monitorear estratégicamente sus negocios actuales y anticiparse para la creación de negocios futuros. Un sistema adecuado para obtener información facilitará este devenir tan complejo, siempre que cuente con la dinámica necesaria para transformar dicha información en conocimientos capaces de generar soluciones diferentes en el día a día.

A su vez la inteligencia tiene su correlato en la estrategia, pues para que los negocios de hoy sean competitivos, necesitan de un continuo flujo de información anticipada sobre una situación de mercado que en pocos días puede cambiar.

De modo que la mayor utilidad en una buena administración de la información es la de estudiar el proceso de evolución de los datos tal que se esté en óptimas condiciones de analizar desvíos iniciales sobre planes propuestos y tomar anticipadamente medidas correctivas.

13. Analizar las consecuencias de cada decisión.

La inteligencia de negocios permite conocer las consecuencias ocurridas similarmente medidas en el pasado. Esto mejora la calidad de las decisiones acercándolas a ser más racionales. Por lo expuesto permite suplir la falta de experiencia del analista, alimentándola con vistas de la información existente y con posibles análisis de sensibilidad.

Puede facilitar el pre-cálculo de comportamientos cercanos a las consecuencias de todas las posibles decisiones aportando racionalidad en muchas de las alternativas consideradas y puede analizar las consecuencias una vez tomada la decisión. Puede dedicar tiempo a investigar decisiones similares tomadas en el pasado y procesar su propia adaptación al nuevo conocimiento generado. La oportuna comunicación de todo lo que rodea sus decisiones favorecerá a toda la organización. Esta difusión de condiciones y consecuencias de su actuación, favorecerá la docilidad del personal que se encuentra en la línea de mando para adquirir esos nuevos conocimientos para aplicarlos a sus áreas de actuación y decisión trabajando con mayor racionalidad.

Adquirir hábitos provenientes de experiencias en forma sistemática, implica un comportamiento repetitivo y decisiones con mayor aseguramiento de objetivos buscados. Esto aumenta su racionalidad, pues el hábito se hace cargo de la integración de los conocimientos emergentes de las decisiones prestando mayor atención a aquellos aspectos diferenciales que deberá considerar en una futura oportunidad, desecharlos por eventuales o revisarlos en caso de haber cambiado las circunstancias.

14. Resultados.

La instalación y puesta en marcha de un sistema, son dos de las últimas etapas del ciclo de vida tradicional de desarrollo de un software. Particularmente, la última etapa requiere de una serie de concurrencias coordinadas que exigen mucha atención. Los datos preexisten a la instalación del producto de Business Intelligence, por lo tanto su uso aporta resultados inmediatamente pues no incorpora nuevos conceptos sino que automatiza mecanismos ya existentes con alta velocidad y sin preocupación por su calidad. No hay necesidad de invertir tiempo en recolectar datos. Apenas instalado, él mismo incorpora los datos habilitando información elaborada con el nivel de detalle deseado. Esto es particularmente útil en relación a la calidad (y poca cantidad) del tiempo disponible por los usuarios de alto nivel gerencial, particularmente cuando se trata de una reciente implementación o para capacitación.

15. Consideraciones sobre el uso de la tecnología.

Mucho se ha discutido sobre la disposición que hacen los gerentes al uso directo de diferentes aplicaciones de software. Es curioso resaltar que muchos empleados hacen uso masivo del mismo software de aplicación que sus autoridades: muchos documentos con un procesador de texto y menos trabajos con plantillas de cálculo. Recíprocamente, las autoridades trabajan mucho el presupuesto con plantillas electrónicas y relativamente menos con documentos de texto para cumplir con sus necesidades. Sin embargo el

personal de nivel subgerencial es mucho más dado al uso de la computadora y hasta lo hace naturalmente. Los gerentes no tienen una disposición similar. A pesar de utilizar el mismo hardware para trabajar, el uso es diferente.

Razones: tal vez porque no tienen la obligación de hacerlo o porque las aplicaciones informáticas específicas no tienen facilidades de acceso o quizás porque sienten perder independencia.

Muchos de ellos no lo saben conscientemente pero trabajan mucho más eficientemente gracias a los productos de tecnologías de la información en trabajos no específicos relacionados con la toma de decisiones. La administración del conocimiento creció y se complicó, por lo que los gerentes deben basarse en métodos más rápidos y más nutridos de información.

No deben dejar de considerarse los aspectos de capacitación y educación, dos cuestiones que en general consumen grandes recursos en costo para los capacitadores como en el escaso y valioso tiempo de los gerentes. En este aspecto es interesante el aporte realizado por las empresas que implementan sistemas de inteligencia de negocios pues no exigen ningún conocimiento técnico previo a adquirir por parte de los gerentes y analistas; las interfaces son muy amigables y la lógica de uso es reducida. Finalmente, los productos de BI ponen a disposición de los que deciden criterios especificados por ellos mismos, por lo que la vinculación casi no requiere educación técnica.

1.4 Medición de las variables de televisión en México

1.4.1 Nielsen IBOPE México

IBOPE son las siglas del Instituto Brasileño de Opinión Pública y Estadística, es un organismo que se encarga de medir televisión, radio, espectáculos, internet, prensa y revistas.

En 2012 Nielsen e IBOPE AGB México unieron su experiencia y trayectoria para complementarse para así brindar a la industria un conocimiento integral del consumidor, que comprende un ciclo en su interacción con los medios y la compra de productos.

Desde su fundación en 1991, IBOPE ha estudiado el comportamiento de las audiencias en México, así como su inherente relación con la actividad publicitaria. Nielsen, por su parte, ha indagado acerca de los hábitos del consumidor mexicano desde 1967. El punto de

encuentro: la investigación rigurosa que día tras día genera información indispensable para saber con certeza ¿Quién? y ¿Cómo es el consumidor?.

Nielsen IBOPE México es el grupo líder en métrica y generación de conocimiento sobre audiencias de medios de comunicación en México, con más de 18 años de experiencia y reconocido prestigio. Esta firma pertenece a Grupo Delphi y a la red IBOPE Internacional AGB Nielsen Media Research con presencia en más de 40 países.

En México, IBOPE mide los ratings basándose en una muestra de varios millones de hogares distribuidos en 28 ciudades de la República Mexicana que albergan más de 50 millones de personas, miden casi el 50% de la población. Para medir el rating de televisión, IBOPE se basa actualmente en el “Establishment Survey”, una encuesta que recopila la información sociodemográfica más relevante a nivel nacional.

Considera rubros como año de adquisición del televisor, tamaño, unidades por hogar, tipo de señal de televisión pagada y lugar físico donde se encuentran los televisores. El reclutamiento aleatorio de los hogares que conforman el panel permite conocer características como sexo, edad, ocupación e integrantes por familia, entre otros datos.

Las asociaciones y organismos a los que pertenece Nielsen IBOPE México son:

- Asociación Mexicana de Agencias de Investigación de Mercado y Opinión Pública (AMAI).
- Consejo de Investigación de Medios (CIM).
- European Society for Opinion and Marketing Research (ESOMAR).
- Asociación Mexicana de Internet (AMIPCI).
- Interactive Advertising Bureau (IAB).

1.4.2 Conceptos

❖ Rating

El rating es la medida del consumo de un programa de TV o de radio, de un bloque horario, de una transmisión publicitaria o de un medio de comunicación, teniendo en cuenta un target (o público objetivo).

Rating es un indicador básico de audiencia que relaciona la cantidad de audiencia ya sean personas u hogares con el tiempo de exposición al medio. En la medición de audiencia de televisión la unidad mínima es el minuto y expresa la población promedio que mira un programa o un canal durante cada minuto de su emisión.

❖ Share

El share es la proporción del consumo de un programa de TV o radio, emisora de radio o canal de TV en relación con el consumo total del medio (encendido). El share indica qué porcentaje de participación tiene cada canal o emisora de radio con respecto al encendido total en un momento determinado, por consiguiente, el share permite la comparación entre competidores.

El share no brinda información sobre la dimensión de la audiencia de un programa, un canal o una radio ya que el share solamente informa la proporción de dicha audiencia con respecto al consumo total del medio (encendido).

❖ Target

Target es público objetivo que se analiza. Cuando se realiza una evaluación de pauta publicitaria el target es el público objetivo al cual está dirigida la comunicación publicitaria. El target se puede segmentar por variables tales como: género, edad, nivel socio económico y zona geográfica.

❖ People Meter

Es el nombre del dispositivo electrónico que se instala en cada uno de los hogares seleccionados como parte de la muestra.

Este microcomputador tiene la capacidad de registrar automáticamente en su memoria el momento en que se enciende el televisor y el canal sintonizado, además de registrar a las personas que se encuentran viendo la televisión. Mide minuto a minuto: encendido de la televisión, selección de canal, tiempo de permanencia en un canal, cambios de canal y personas que ven la televisión.

Gracias a que el proceso de transmisión de datos de audiencia se realiza por radio frecuencia, la central de procesos recibe esta información en forma simultánea, es decir, cuando en cada hogar de la muestra se producen cambios en el comportamiento de teleaudiencia, en nuestras oficinas el registro es inmediato, calculándose las estimaciones de rating y encendido On-Line para cada minuto del día en forma continua, todos los días del año. Esta información es enviada en paralelo a los distintos clientes que contratan este servicio.

Capítulo II

DEFINICIÓN DEL SISTEMA

2.1 Definición del problema

Hemos notado que la recolección de la información en grandes cantidades que un usuario final llega a necesitar para realizar un análisis o reporte informativo de sus programaciones televisivas diarias, suele ser un proceso demasiado complejo, tardado y requiere de mucho personal dedicado exclusivamente a esas tareas.

Actualmente existen aplicaciones con costo por licencia de uso que nos permiten acceder a gran nivel de detalle a su base de datos y así contar con información de audiencia y programación en televisión. Normalmente los usuarios de estas aplicaciones son personas experimentadas con ésta y otras herramientas como Microsoft Excel, estos usuarios realizan parte de su análisis generando reportes informativos, pero se enfrentan con obstáculos como:

- Las capacidades de presentación de la información de las aplicaciones son muy limitadas y por ello deben hacer uso de Microsoft Office para complementar dicha presentación.
- El análisis de la información histórica de la empresa, por su tamaño llega a ser excesivamente lento y en ocasiones hasta riesgoso ya que se puede saturar la memoria RAM y muchas veces se rebasan los límites de capacidad de las hojas de cálculo.

Para complementar estos análisis y generación de reportes los usuarios deben crear procesos con macros y así superar la ausencia de la escalabilidad en Microsoft Excel en combinación con el programa de IBOPE.

Con este sistema pretendemos proporcionar una herramienta que enriquezca las facultades de análisis y presentación de la información de audiencia de las televisoras mexicanas. Permitiendo a los usuarios finales el manejo fácil de datos y la presentación de la información que necesiten exponer ante directivos de las empresas donde colaboren.

La plataforma Qlikview es una herramienta que satisface las necesidades de la problemática identificada ya que da respuesta a las necesidades de los usuarios que quieren tener una analítica intuitiva y de rápido despliegue.

En la actualidad la transición es hacia plataformas que puedan implementarse en poco tiempo y que puedan ser usadas para encontrar los conocimientos rápidamente sin estar centralizados los datos en una sola área. Respondiendo a las necesidades cambiantes de una organización.

Qlikview permite consolidar información y explorar datos de forma interactiva, ofrece cientos de tablas y objetos gráficos posibles y sus variedades: cuadros de lista para explorar y seleccionar dimensiones de los datos, cuadros de estadísticas, gráficos y muchos otros elementos visuales en la interfaz. Permite crear tableros de control y analizar distintas fuentes de datos obteniendo una visualización general de la televisora. No importa la cantidad ni el tipo de las fuentes de información. Ofrece una verdadera simplicidad que aporta respuestas al instante para todo tipo de usuarios, no sólo unos cuantos expertos, despierta su curiosidad y creatividad. En la Figura 2.1 se observa como Qlikview puede extraer y transformar datos de una o más fuentes (por ejemplo, ERP, texto, Excel, XML).

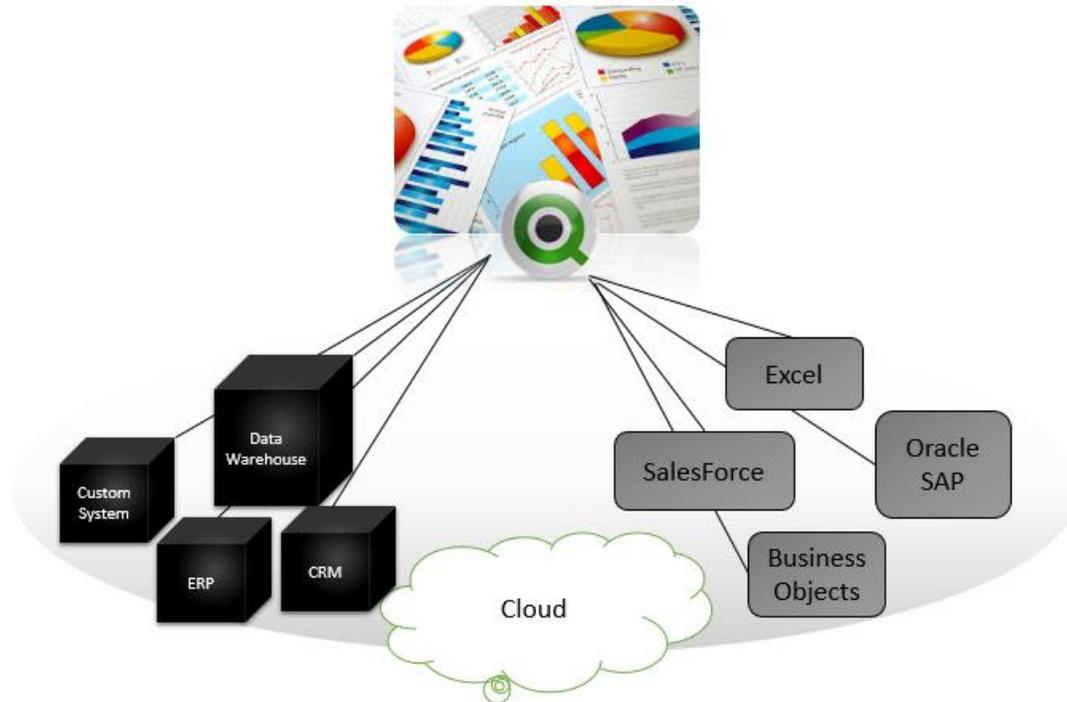


Figura 2.1. Extracción y transformación de datos. ¹

1. Diseño tomada de la Pág. <http://qlikview.6si.com.mx/como-comprar/#.VQSw2LI0zIX>

Nuestro sistema permitirá a los usuarios explotar a profundidad la herramienta con un mínimo apoyo de las áreas de desarrollo y sistemas. Permitiendo a las autoridades de las televisoras contar con la información necesaria para detectar tendencias en la transmisión de contenido de la competencia y así poder tomar las medidas necesarias para reaccionar y/o anticiparse a los movimientos planeados por la competencia.

2.2 Análisis de requerimientos de datos

En este apartado del análisis de requerimientos de datos es importante dimensionar correctamente el sistema para garantizar su rendimiento. La plataforma integra múltiples fuentes de datos que pueden ser: SAP, Sales Force, Oracle, SQL, Business Objects, Excel, txt, www, cloud, etc., cuyo acceso depende de los requerimientos de cada fuente.

Para la construcción del sistema de apoyo en la planeación de la programación en televisoras mexicanas y con fines demostrativos sólo se utilizarán archivos en formato de Microsoft Excel (.xlsx) y toda la información utilizada en el sistema es inventada.

El sistema permitirá el análisis considerando lo siguiente:

- Los canales de televisión considerados son:
 - 2002 y 2005 de la empresa Teleprisa.
 - 2007 y 2013 de la empresa Tvasecas.
 - 2001 de la organización TVFI.
 - 2006 de la organización TVESCOM.

- Fecha de transmisión de un programa.
- Duración de un programa.
- Targets considerados:
 - Género: Hombre o mujer.
 - Edad:
 - Niños de 4 a 12.
 - Jóvenes de 13 a 18.
 - Jóvenes de 19 a 29.
 - Adultos de 30 a 44.
 - Adultos de 45 a 54.
 - Adultos de 55+.

- Niveles socioeconómicos (NSE):
 - Pr ABC+ clase alta.
 - Pr C clase media.
 - Pr D+ clase baja.
 - Pr DE+ austero.
- Tipo de servicio de televisión:
 - Pr c TV de paga.
 - Pr s TV de paga.
- Variables de audiencia:
 - Share (Shr%).
 - Rating (Rat%).
 - Encendidos (Tvr%).
- Regiones:
 - DF.
 - Monterrey.
 - Guadalajara.

Para el análisis de requerimientos de datos nos apoyamos en las siguientes tablas:

- 1) Requerimientos de datos.
- 2) Reglas de negocio.
- 3) Cálculos (fórmulas y/o funciones).
- 4) Dimensión vs métricas.

Tabla de requerimientos de datos:

El objetivo de esta tabla 2.1 es listar todos los datos a utilizar en el sistema, especificando de forma clara si se trata de dimensión, métrica o cálculo. Es importante definir la fuente origen del dato con su nombre técnico (Oracle, etc.) y anotar de forma clara la descripción del dato.

Dato	Campo	Tipo	Tabla	Fuente	Descripción
Rating	Rat%	Métrica	DF, Gda, Mty	(IBOPE) Archivo Excel Formato de archivo (hora a hora) yyyyymmdd Formato de archivo (Programación) yyyyymmdd	Porcentaje de personas que estuvieron expuestas a la televisión en un momento determinado, referido al universo.
Share	Shr%	Métrica	DF, Gda, Mty	(IBOPE) Archivo Excel Formato de archivo (hora a hora) yyyyymmdd Formato de archivo (Programación) yyyyymmdd	Es la participación que tiene un canal sobre el total.
Encendidos	Tvr%	Métrica	DF, Gda, Mty	(IBOPE) Archivo Excel Formato de archivo (hora a hora) yyyyymmdd Formato de archivo (Programación) yyyyymmdd	Porcentaje de telespectadores medios del total de televisiones en un momento determinado.

Tabla 2.1. Tabla requerimientos de datos.

Dato	Campo	Tipo	Tabla	Fuente	Descripción
Canal	Canal	Dimensión	DF, Gda, Mty	(IBOPE) Archivo Excel Formato de archivo (hora a hora) yyyyymmdd Formato de archivo (Programación) yyyyymmdd	Canal de televisión.
Hora	Hora	Dimensión	DF, Gda, Mty	(IBOPE) Archivo Excel Formato de archivo (hora a hora) yyyyymmdd Formato de archivo (Programación) yyyyymmdd	Unidad de tiempo.
Región	Región	Dimensión	DF, Gda, Mty	(IBOPE) Archivo Excel Formato de archivo (hora a hora) yyyyymmdd Formato de archivo (Programación) yyyyymmdd	Localización del público objetivo (target).
Título	Programa	Dimensión	DF, Gda, Mty	(IBOPE) Archivo Excel Formato de archivo (hora a hora) yyyyymmdd Formato de archivo (Programación) yyyyymmdd	Nombre programa.

Tabla 2.1. (Continuación) Tabla requerimientos de datos.

Dato	Campo	Tipo	Tabla	Fuente	Descripción
Sexo	Género (Tipología)	Dimensión	DF, Gda, Mty	(IBOPE) Archivo Excel Formato de archivo (hora a hora) yyyymmdd Formato de archivo (Programación) yyyymmdd	Caricaturas, cómicos, concurso, cultural, etc.
Fecha	Fecha	Dimensión	DF, Gda, Mty	(IBOPE) Archivo Excel Formato de archivo (hora a hora) yyyymmdd Formato de archivo (Programación) yyyymmdd	Fecha de transmisión del programa.
Target	Target	Dimensión	DF, Gda, Mty	(IBOPE) Archivo Excel Formato de archivo (hora a hora) yyyymmdd Formato de archivo (Programación) yyyymmdd	Público objetivo.

Tabla 2.1. (Continuación) Tabla requerimientos de datos.

Tabla reglas de negocio:

El objetivo de esta tabla 2.2 es establecer de forma clara y concisa la regla que define la dimensión y/o métrica y/o cálculo, si existe la participación de datos (dimensiones establecidas) deben ser anotados con el mismo nombre utilizado en la hoja “Requerimientos de datos”.

Nota: En las variables de los archivos de programación y hora a hora, no se podrán realizar agrupaciones, tampoco operaciones de suma, resta, promedio. NA= No aplica

Consecutivo	Dato	Regla de negocio	Excepción	Inclusión	Observaciones	
NA	NA	NA	NA	NA	NA	NA
NA	NA	NA	NA	NA	NA	NA
NA	NA	NA	NA	NA	NA	NA
NA	NA	NA	NA	NA	NA	NA
NA	NA	NA	NA	NA	NA	NA
NA	NA	NA	NA	NA	NA	NA

Tabla 2.2. Tabla regla de negocios.

Tabla cálculos (fórmulas y/o funciones).

El objetivo de esta tabla 2.3 es describir los elementos y operaciones que componen la fórmula, se recomienda el uso de colores para distinguir los componentes diferenciando entre **métrica (verde)**, **cálculo (rojo)** y **variables (azul)**.

Nota: NA = No aplica

Dato	Fórmula
Duración	Duración = PrgTiempoFinF – PrgTiempoInicioF
NA	NA

Tabla 2.3. Tabla cálculos (fórmulas y /o funciones).

Tabla dimensión vs métricas.

El objetivo de esta tabla 2.4 es definir el nivel de detalle de la métrica respecto a la dimensión, mediante el cruce dimensión vs métrica. En resumen, se determina la viabilidad del llegar o no a un detalle solicitado.

		Identificación de correspondencias dimensiones						
Fuente de datos		Canal	Hora	Dominio	Programa	Género (Tipología)	Fecha	Target
Programación	Rat%	✓	☐	✓	✓	✓	✓	✓
	Shr%	✓	☐	✓	✓	✓	✓	✓
	Tvr%	✓	☐	✓	✓	✓	✓	✓
Hora a hora	Rat%	✓	☐	✓	☐	☐	✓	✓
	Tvr %	✓	☐	✓	☐	☐	✓	✓

Tabla 2.4. Dimensión vs métrica.

2.3 Análisis de requerimientos funcionales

El objetivo es definir la funcionalidad de la aplicación, agrupando los objetos por requerimiento de negocio para diseñar la interfaz que responda a las necesidades y alcances planteados. Así como definir los elementos de la aplicación que permitirán al usuario explotar el poder de análisis y simplicidad del sistema.

Para el diseño de nuestro sistema consideramos que éste debe responder a las siguientes preguntas que tienen que ver con las necesidades de operación de las televisoras mexicanas:

- 1.- Se requiere conocer el valor de las variables de audiencia: Share, rating y encendidos de un programa seleccionado.
- 2.- Es necesario conocer también el valor de la variable de audiencia de un canal en comparación con la competencia.
- 3.- ¿Cuál ha sido el comportamiento del valor de la variable seleccionada por el usuario durante sus diferentes fechas de transmisión?
- 4.- Analizar las variables de audiencia de acuerdo a su género, nivel socio económico, tipo de servicio de televisión y sus diferentes edades.

El sistema se alimentará de archivos en formato de Microsoft Office Excel utilizando los campos mencionados en el análisis de requerimiento de datos. Se debe dimensionar adecuadamente para cubrir la mayoría de los reportes cotidianos solicitados por las autoridades y eliminando por completo (o en la mayor medida posible) la intervención humana en estas tareas repetitivas permitiendo con ello que los empleados se dediquen al análisis y no al procesamiento manual de la información.

Nuestro sistema propone una estructura de cuatro pestañas con el detalle siguiente:

- 1.- Una primer pantalla “BIENVENIDO” que muestra el logo de la institución, la fecha actual y la versión del sistema. Ver figura 2.2.



Figura 2.2. Pestaña "BIENVENIDO".

2.- Una segunda pestaña nombrada "ANÁLISIS DE CANALES" en donde se podrán analizar las variables de un canal con respecto a su competencia, la cantidad de programas transmitidos en determinada fecha, así como los valores alcanzados por dichas variables. Contará con cuatro objetos que nos permitirán seleccionar parámetros para filtrar la información y mostrarla en dos gráficos "A" y "B". El gráfico "A" mostrará la cantidad, nombre y la duración de los programas transmitidos en los canales seleccionados por el usuario. El gráfico "B" al estar vinculado mostrará mediante una gráfica de líneas los valores de las variables que seleccione el usuario, permitiendo con ello conocer el comportamiento de esta variable con respecto al valor de la misma en el canal de la competencia. Ver figura 2.3.

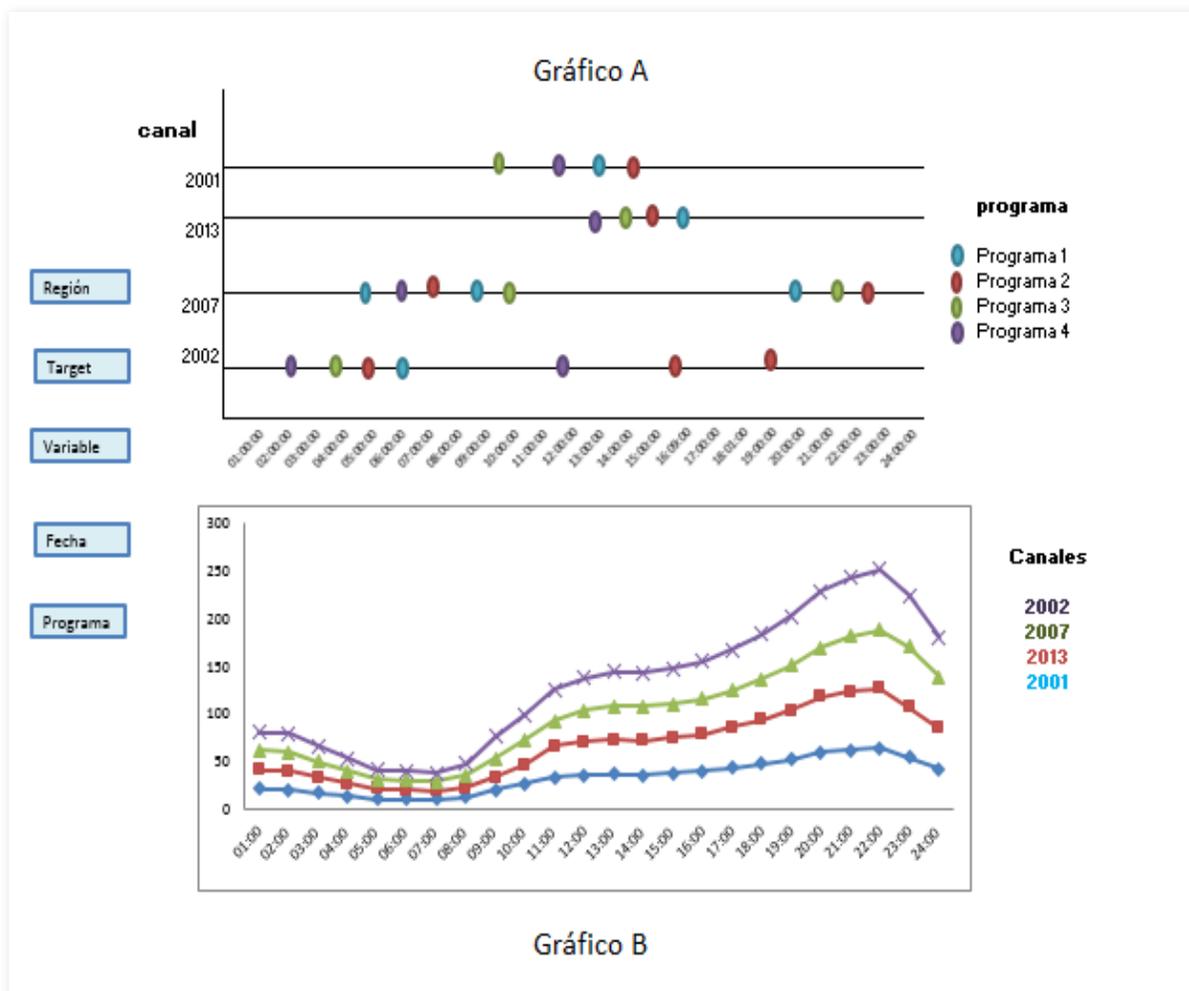


Figura 2.3. Pestaña “ANÁLISIS DE CANALES”.

3.- Una tercera pestaña nombrada “ANÁLISIS DE LA AUDIENCIA” que contará con cuatro objetos de selección múltiple con los que seleccionaremos los parámetros para filtrar la información y así mostrarla en cuatro objetos gráficos, éstos permitirán realizar un comparativo entre dos variables de audiencia seleccionadas por un usuario para un determinado programa.

En los objetos gráficos podremos visualizar la información de acuerdo a los siguientes perfiles:

Gráfico 1 con perfil de tipo de servicio de televisión: mostrará el valor de las variables comparando la audiencia entre personas que cuentan con televisión de paga y personas que no cuentan con televisión de paga. Ver figura 2.4.

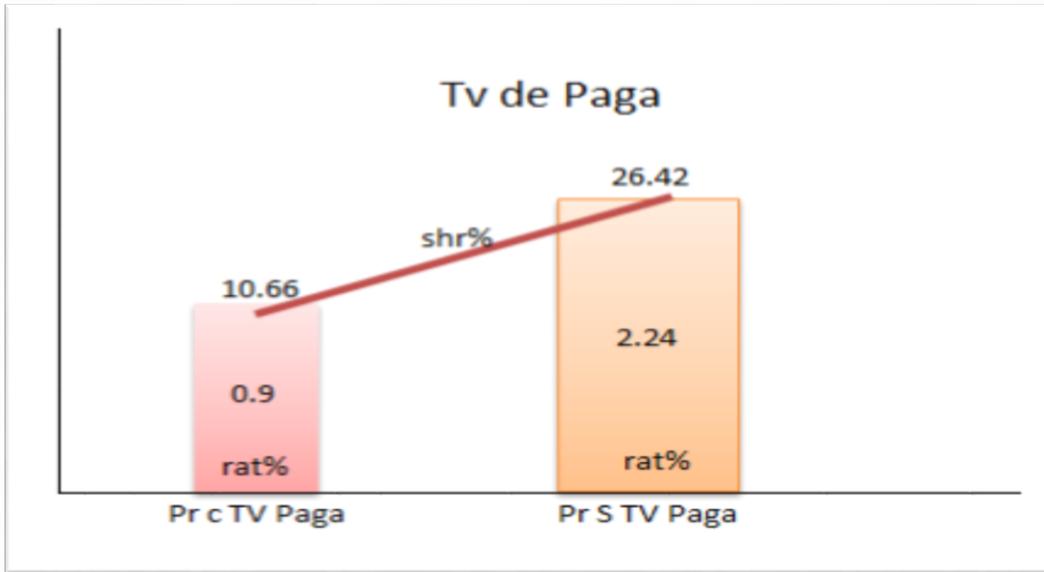


Figura 2.4. Perfil de Tv de paga.

Gráfico 2 con perfil de nivel socio económico (NSE) ver figura 2.5: mostrará el valor de las variables de audiencia de acuerdo a su nivel socio económico siendo:

Nivel socio económico	Clase económica
Pr ABC+	Alta
Pr C	Media
Pr D+	Baja
Pr DE	Austero

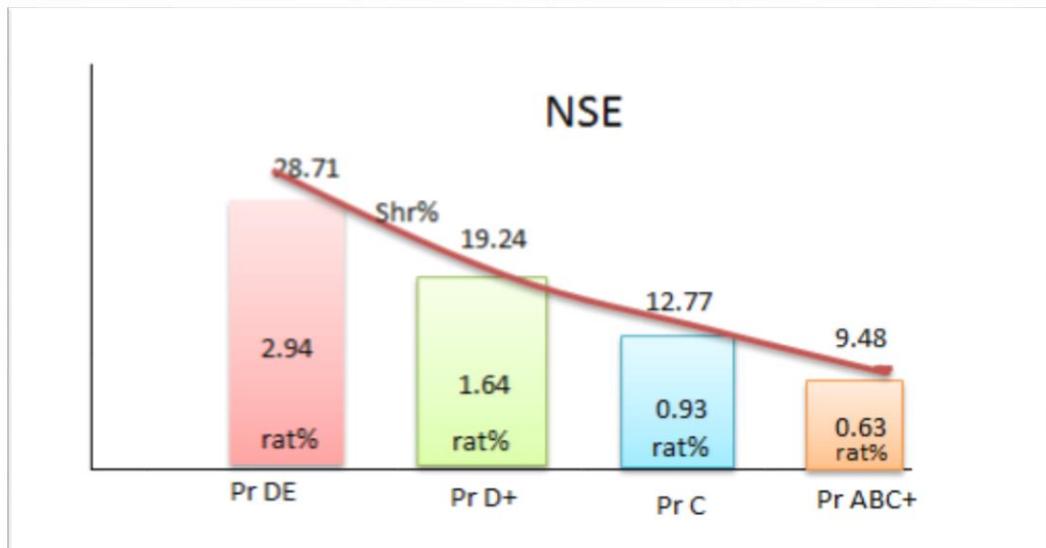


Figura 2.5. Perfil de nivel socioeconómico.

Gráfico 3 con perfil de género: mostrará los valores de las variables para comparar audiencia entre hombres y mujeres. Ver figura 2.6.

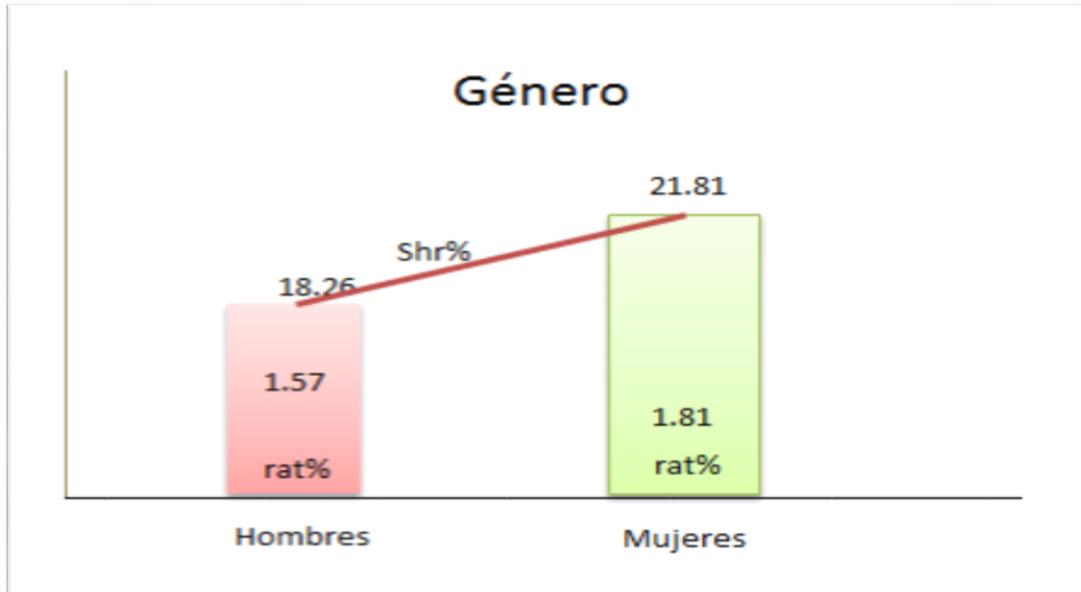


Figura 2.6. Perfil de género.

Gráfico 4 perfil de edad: refleja los valores de las variables seleccionadas por el usuario para cada rango de edades. Ver figura 2.7.

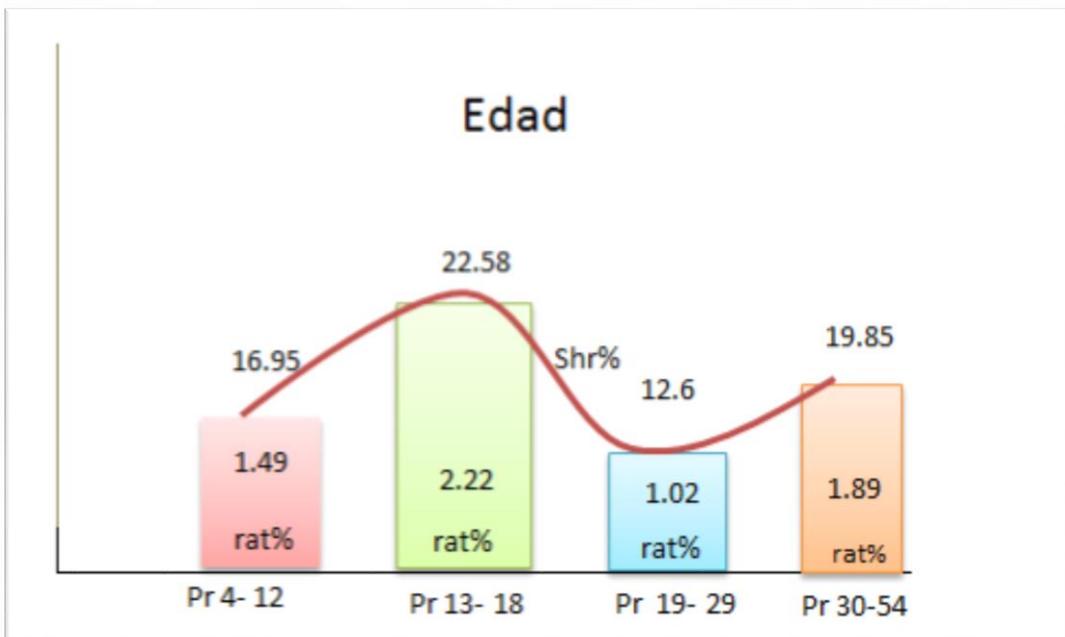


Figura 2.7. Perfil de edades.

4.- Una cuarta pestaña llamada “ANÁLISIS DE UN PROGRAMA” que contará con cuatro objetos para seleccionar parámetros que filtrarán la información y así poder analizar el comportamiento de una variable de audiencia para un programa determinado en sus diferentes fechas de transmisión. Ver figura 2.8.

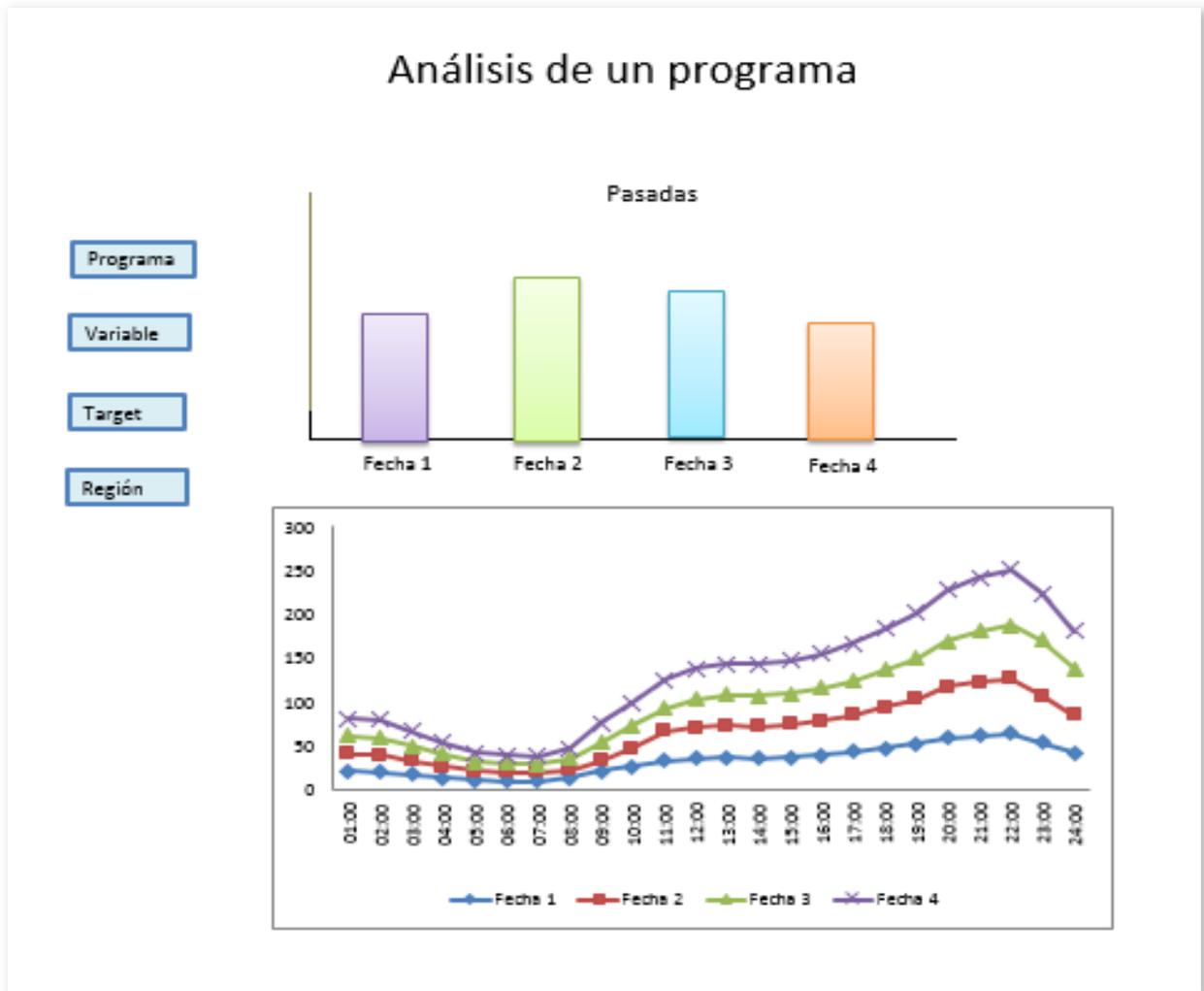


Figura 2.8. Pestaña “ANÁLISIS DE UN PROGRAMA”.

Capítulo III

DESARROLLO DE LA APLICACIÓN

3.1. Extracción de datos

Consiste en extraer los datos desde las fuentes de información que se van a utilizar en el sistema, con la posibilidad de combinar varias de éstas y la finalidad de contar con una base de datos homologada y en formato de Qlikview.

Para ello se utiliza el software Qlikview que descargamos gratuitamente del sitio www.qlikview.com y se realiza la instalación en nuestro equipo de cómputo que fungirá el papel de servidor de prueba como se explica a continuación:

Una vez descargado el archivo de instalación, se debe ubicar y hacer doble click en el instalador. Aparecerá una pantalla de extracción del archivo con la extensión “.msi” como se muestra en la figura.3.1.



Figura 3.1. Pantalla de extracción del archivo Qlikview.msi.

Enseguida aparecerá el wizard de instalación y dar click en: **“Next”**. Ver figura 3.2.



Figura 3.2. Pantalla wizard de Qlikview.

En la pestaña “Select Region” debe seleccionar “Other” ya que México no aparece como opción y dar click en: **“Next”**. Ver figura 3.3.

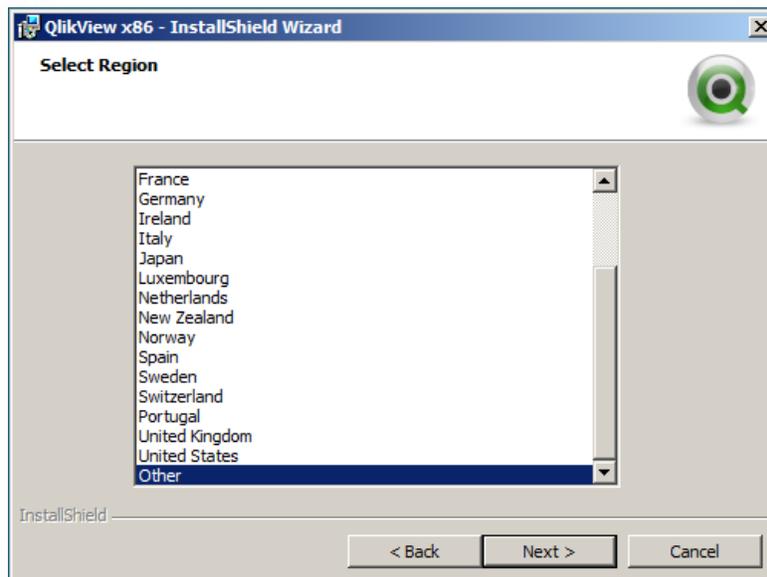


Figura 3.3. Pantalla de selección de región de Qlikview.

Leer el acuerdo de licencia y aceptar los términos. Si fuera necesario, imprimir el acuerdo de licencia y dar click en: **“Next”**. Ver figura 3.4.



Figura 3.4. Pantalla de Licencia de Qlikview.

Ingresar el nombre de usuario y organización. Dar click en: **“Next”**. Ver figura 3.5.

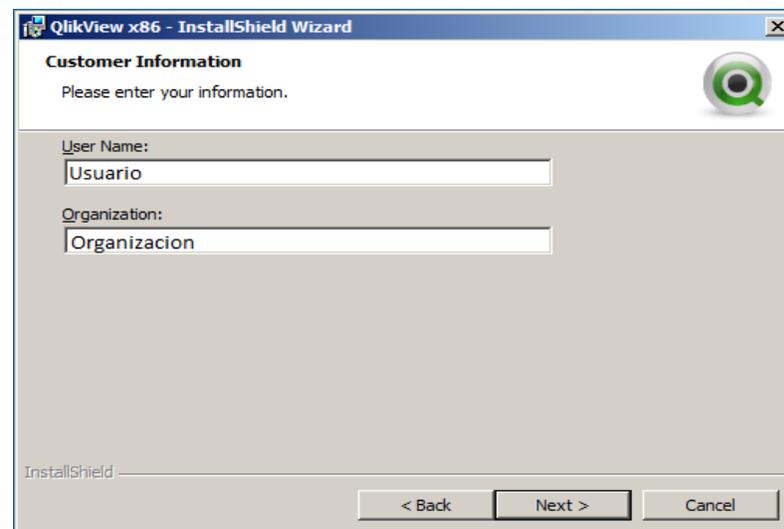


Figura 3.5. Pantalla de ingreso de usuario y organización de Qlikview.

Si se desea instalar Qlikview en una carpeta diferente, es necesario cambiar la ubicación de la carpeta destino. Dar click en “**Next**”. Ver figura 3.6.



Figura 3.6. Pantalla de selección de ubicación de Qlikview.

El wizard permitirá escoger entre la instalación “Complete” o “Custom”. Seleccionar “Complete” y dar click en: “**Next**”. Ver figura 3.7.

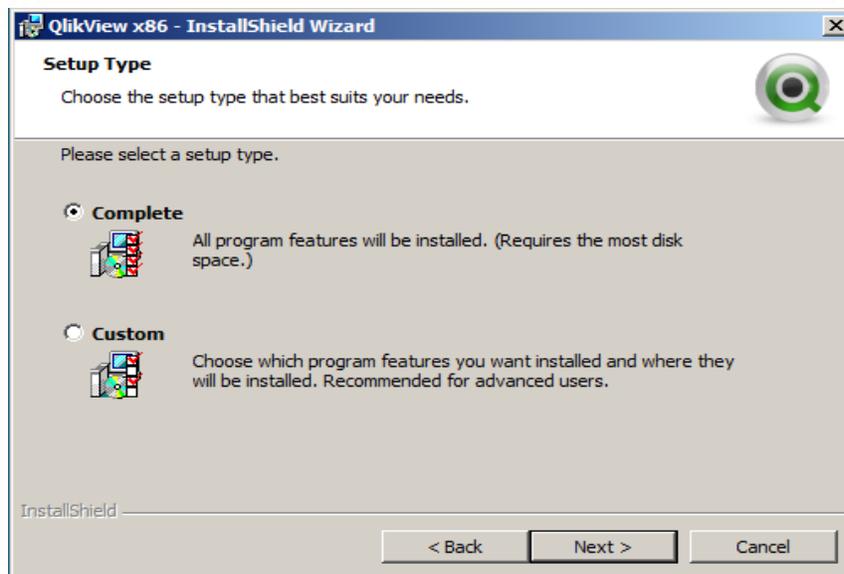


Figura 3.7. Pantalla de instalación completa o cliente de Qlikview.

Instalación de Qlikview 11 en proceso. Se puede cancelar la instalación en el momento deseado. Ver figura 3.8.

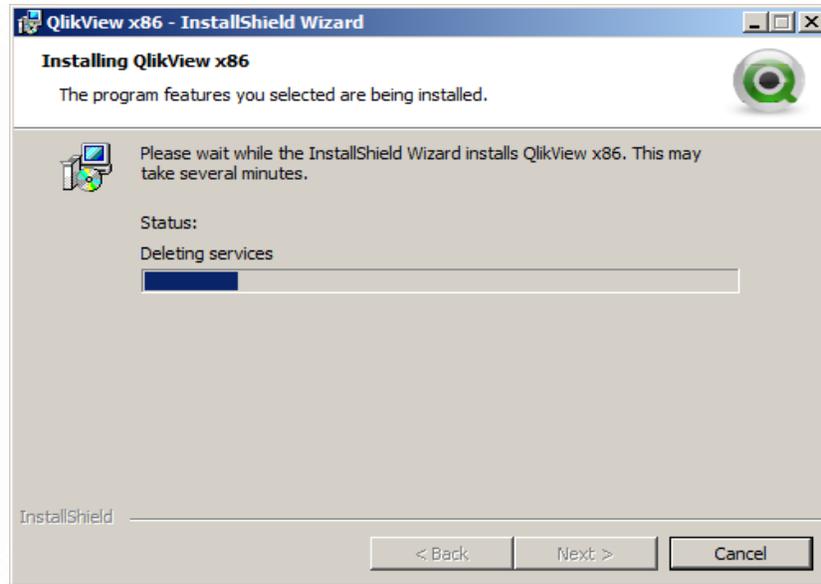


Figura 3.8. Pantalla de instalación de Qlikview.

Instalación de Qlikview 11 concluida. Dar click en: “**Finish**”. Activar el recuadro “Show the Windows Installer log” para ver el log de instalación. Ver figura 3.9.

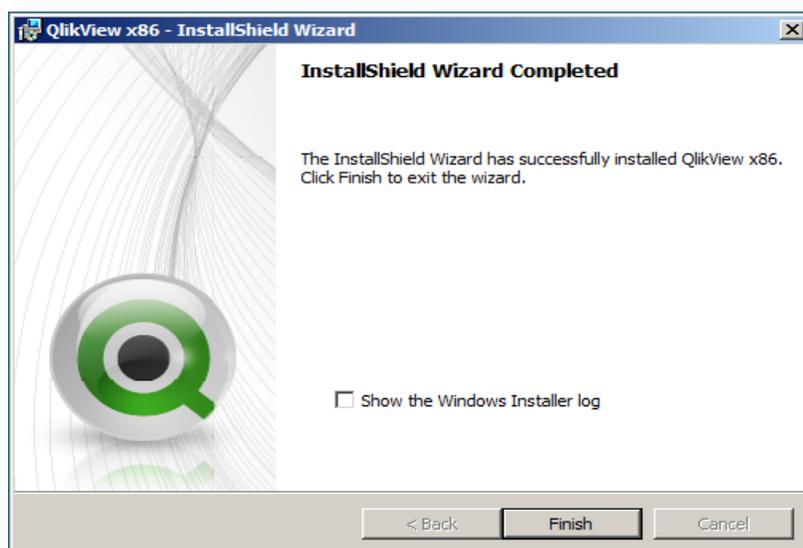


Figura 3.9. Pantalla de terminación de la instalación de Qlikview.

Abrir la aplicación Qlikview 11 y diríjase a la ventana “Ayuda” y enseguida seleccionar “Acerca de Qlikview” para ver que versión de Qlikview fue instalada. Ver figura 3.10.



Figura 3.10. Ventana de ayuda de Qlikview.

Para dar de alta una licencia, se debe seleccionar “Configuración” enseguida “Preferencias de usuario” y en el recuadro que se abre, seleccionar la pestaña “Licencia”, dar click en “Cambiar” y aparecerá una ventana en donde se debe ingresar la clave de licencia y el número de control. Ver figura 3.11.

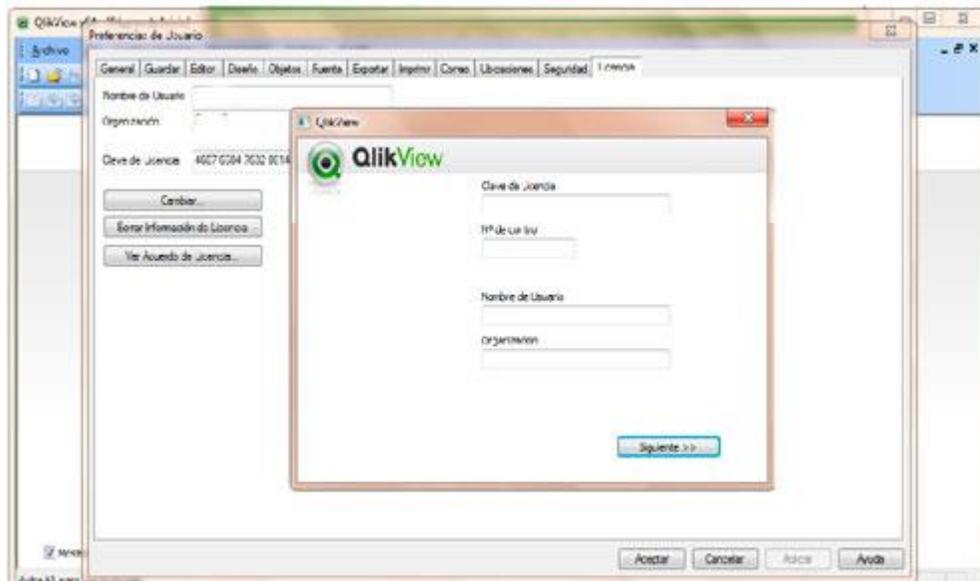


Figura.3.11. Pantalla de preferencias de usuarios de Qlikview.

Ya instalado el programa, éste se puede utilizar indefinidamente. Es importante mencionar que si se cuenta con Qlikview en su versión “Personal Edition” (descargada del sitio web oficial de Qlikview) no podrá visualizar los trabajos realizados en otros equipos, para ello debe contar con una licencia.

Con la herramienta Qlikview se tiene conectividad de datos para múltiples fuentes:

- Archivos: se utiliza el asistente de datos incorporado en Qlikview para conectar con fuente de datos como hojas de cálculo, XML y páginas web.
- Base de datos: se extraen datos mediante el conector ODBC o OLEDB o conectores nativos Qliktech proveedores de base de datos.
- Consultas a otros sistemas de información: se conectan a otros sistemas definiendo cada uno sus parámetros.
- Fuentes de datos personalizadas: tienen su propio conector a la base de datos.

Una vez extraídos, los datos se convierten a formato “.qvd” (Qlikview data) lo que permite su fácil manejo y asegura la integridad en la información (los datos origen no se modifican). Debido a la gran manipulación de información es necesario revisar e implementar algunas prácticas para lograr mejores y más limpios diseños buscando que el sistema mantenga consistencia sin datos innecesarios; muchas veces empezamos a generar archivos, algunos de ellos pueden no estar relacionados entre sí y se recargan de forma separada en determinado momento, con información redundante que comienza a hacerse insostenible su actualización por el tiempo de recarga de nuevos datos.

Las mejores prácticas que utilizamos en nuestro sistema son:

1. Tener bien definido el objetivo de nuestros documentos.
2. Trabajar con archivos en formato qvd que nos ofrecen las siguientes ventajas:
 - a) Velocidad (entre 10 y 100 veces más rápida la recarga de un archivo qvd que un equivalente en otro formato).
 - b) La estructura de los campos es única y hay un único punto de mantenimiento.
3. Invertir más en el script de carga y menos en formulación en los objetos, con ello se gana velocidad frente al usuario teniendo el dato ya pre-calculado.
4. Utilizar nombres claros que permitan el fácil mantenimiento del sistema y comprensión de los campos así como de las variables.
5. El aspecto del sistema debe ser muy gráfico para atraer la atención del usuario.
6. Es muy importante generar respaldo de todo.

La información que se maneja (y se genera) en el sistema se almacena con la estructura de directorios NTFS que se muestra en la figura 3.12 siguiente:

Estructura de directorios:

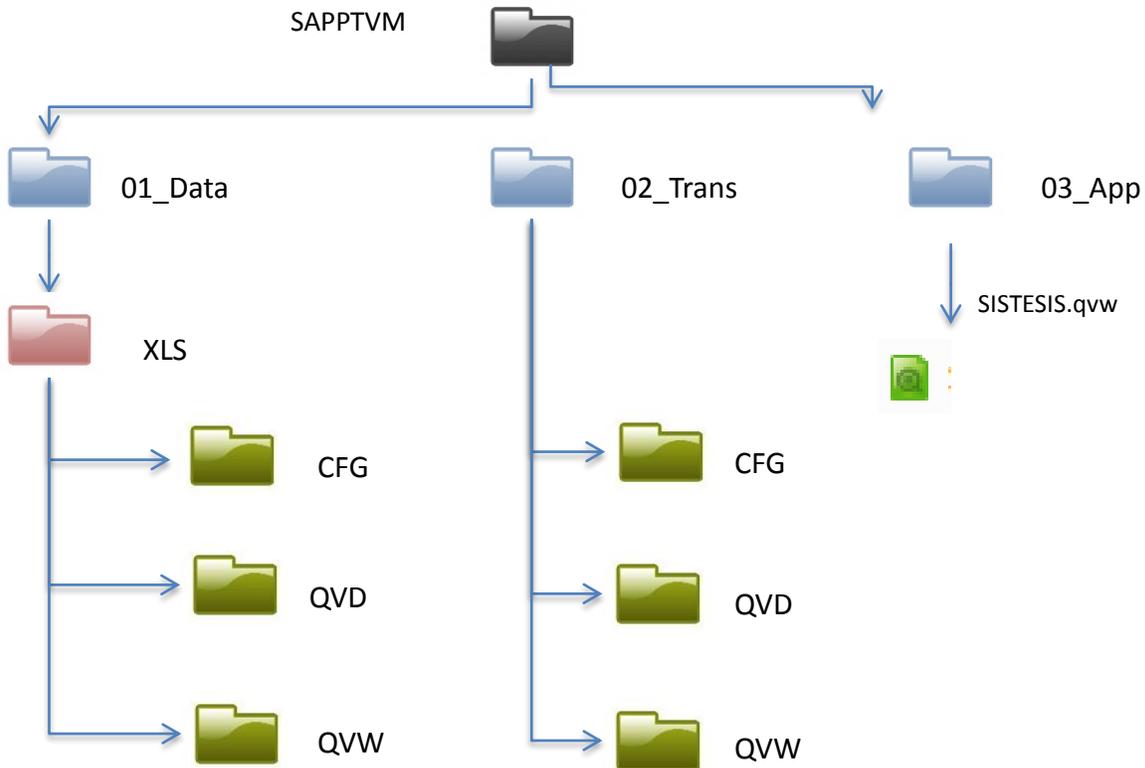


Figura 3.12. Estructura de directorios.

01_DATA: contiene los archivos fuente y los generados en formato qvd.

XLS

CFG : Contiene los archivos fuente.

QVD: Contiene los archivos qvd (Qlikview Data) generados.

QVW: Contiene el archivo ejecutable para la extracción de datos.

02_TRANS: contiene archivos extraídos de 01_DATA y QVD generados para el modelo asociativo.

CFG: Puede contener archivos externos necesarios para la transformación.

QVD: Contiene los archivos qvd (Qlikview Data) generados.

QVW: Contiene el archivo ejecutable para la transformación de datos.

03_APP

Contiene el archivo “.qvw” (formato Qlikview Worksheet) ejecutable del sistema.

El proceso de ejecución del sistema se muestra en la figura 3.13.

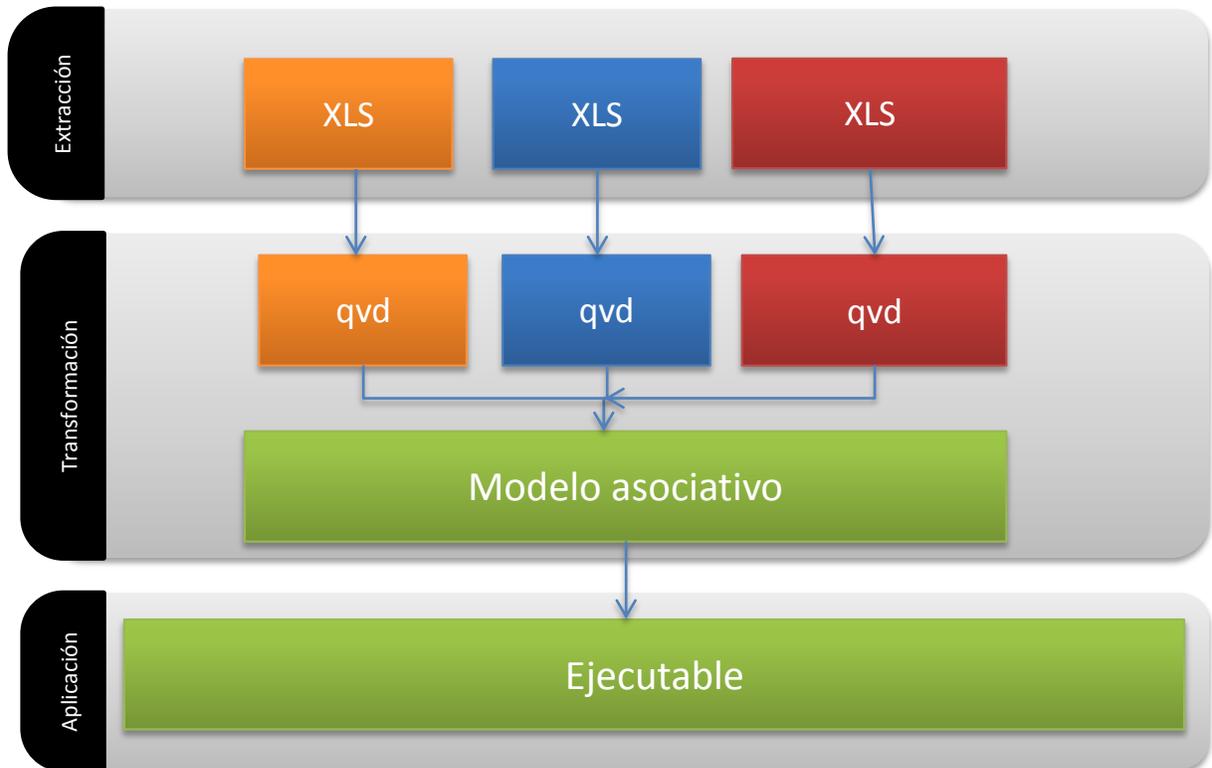


Figura 3.13 Proceso de ejecución del sistema.

Para nuestro sistema se utilizan archivos de hoja de cálculo en la versión de Microsoft Office 2010 que deben contar con las estructuras que se describen a continuación:

Archivo primero (hora por hora):

Estructura: los documentos origen deben contener la siguiente estructura, es importante no modificarla porque se corre el riesgo de que la carga de la información no se ejecute correctamente en la base de datos del sistema y los datos no sean visibles. Ver figura 3.14.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	
1																
2		A) Fecha: La fecha debe estar en formato. DDDMMYYYY en la celda A1							E) Var+target: Cada columna muestra el valor de cada target.							
3																
4		B) Canales: Los canales identifican el nombre del canal a proyectar					Fecha	var+target	var+target	var+target	var+target	var+target	var+target			
5						Canal 1										
6						Hora 1										
7						Hora 2										
8		C) Horas: Las horas deben estar con el formato 01:00-02:00					Hora 3									
9															
10						Hora 22										
11						Hora 23										
12						Hora 24										
13						Canal 2										
14						Hora 1										
15						Hora 2										
16						Hora 3										
17															
18						Hora 22										
19						Hora 23										
20						Hora 24										
21						Canal 3										
22						Hora 1										
23						Hora 2										
24						Hora 3										
25															
26						Hora 22										
27						Hora 23										
28						Hora 24										
29															
30						Canal 6										
31						Hora 1										
32						Hora 2										
33						Hora 3										
34															
35						Hora 22										
36						Hora 23										
37						Hora 24										
38						HOJA 1	HOJA 2	HOJA 3								
39																
40																

Figura 3.14. Estructura de archivos .xlsx hora por hora.

Archivo segundo (programación):

Estructura: en la siguiente imagen se muestra la estructura que deben tener los archivos de programación, es importante verificar que cada archivo contenga las tres regiones en sus hojas correspondientes. Ver figura 3.15.

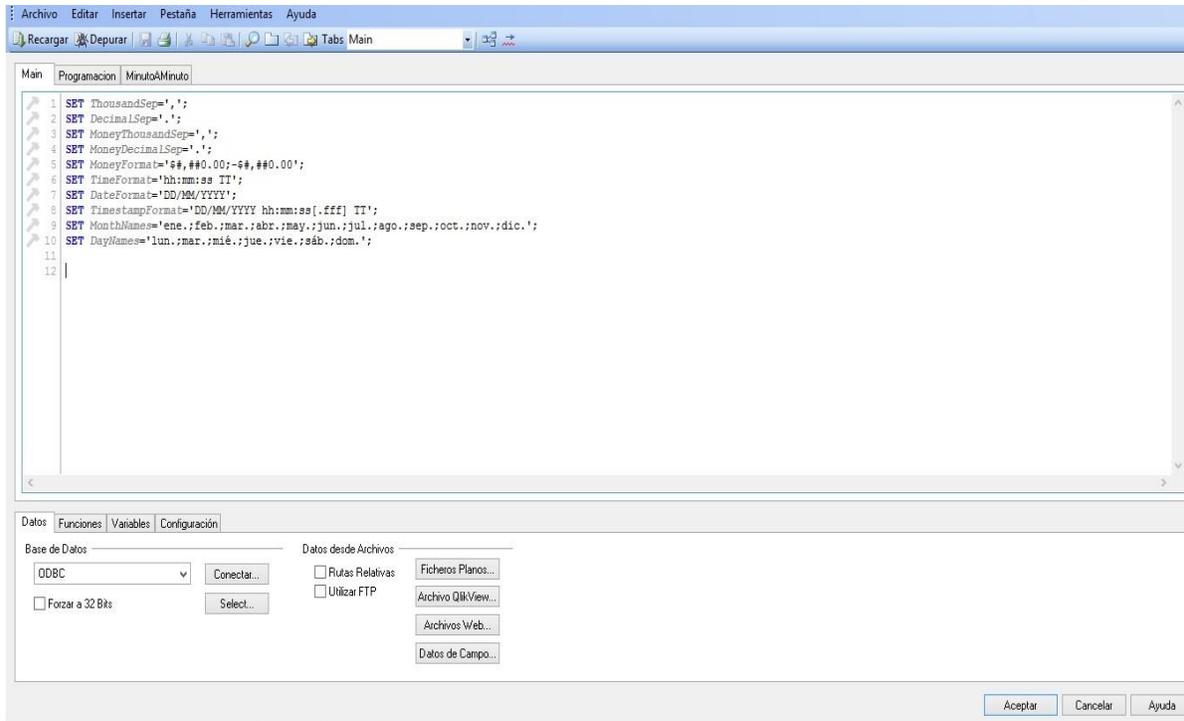


Figura.3.16. Editor de script.

y en consecuencia se generarán los archivos en formatos .qvd y .qvw que servirán en cada etapa posterior (transformación y aplicación) de la construcción de nuestro sistema.

Creamos el archivo “ExtraccionIBOPE5.qvw” el cual contiene el código que permite leer por primera vez los archivos y en un futuro solo se incrementará lo existente.

Dicho código generado para este proceso de extracción de datos se puede consultar en el **ANEXO A** del presente documento.

Una vez ejecutado el archivo “ExtraccionIBOPE5.qvw” nos aparece la imagen mostrada en la figura 3.17 que indica que nuestro proceso de extracción resultó exitoso.

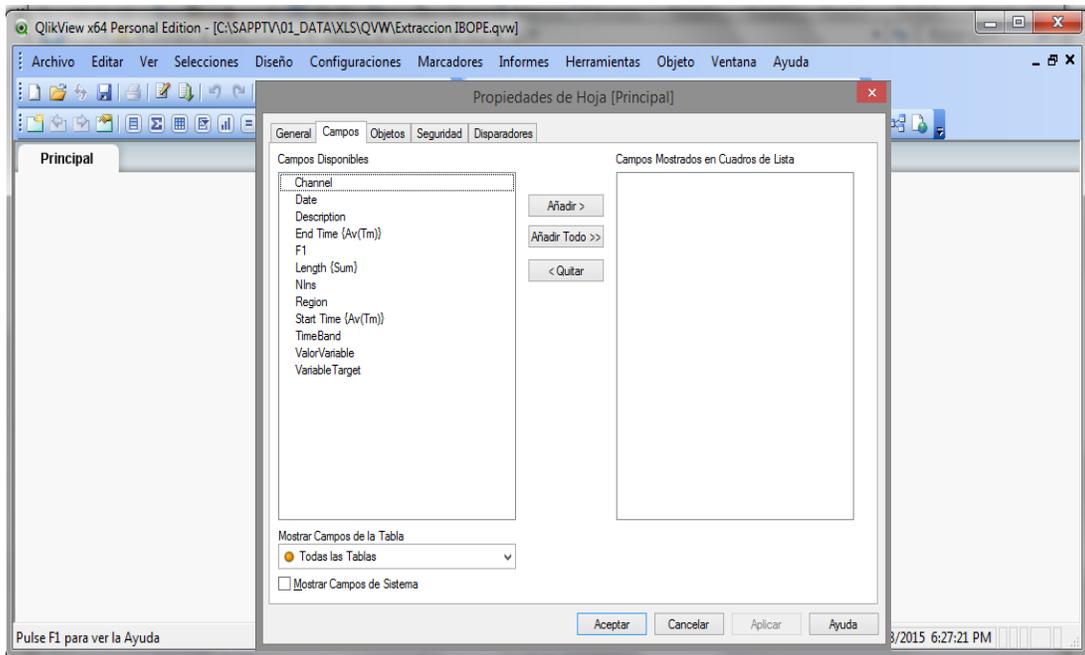


Figura 3.17. Script ejecutado de forma exitosa.

3.2. Transformación

La fase de transformación aplica una serie de reglas de negocio o funciones sobre los datos extraídos para convertirlos en datos transformados que serán cargados en la aplicación. Algunas fuentes de datos requerirán de una pequeña manipulación de los datos. No obstante en otro caso puede ser necesario aplicar alguna de las siguientes transformaciones:

- Seleccionar sólo ciertas columnas para su carga (por ejemplo, que las columnas con valores nulos no se carguen).
- Traducir códigos (por ejemplo, si la fuente almacena una “H” para hombre y “M” para mujer pero el destino tiene que guardar “1” para hombre y “2” para mujer).
- Codificar valores libres (por ejemplo, convertir “hombre” en “H” o “Sr” en “1”).
- Obtener nuevos valores calculados (por ejemplo, duración = hr fin – hr inicio).
- Unir datos de múltiples fuentes (por ejemplo, búsquedas, combinaciones, etc.).
- Calcular totales de múltiples filas de datos (por ejemplo, ventas totales de cada región).
- Transponer o pivotar (girando múltiples columnas en filas o viceversa).
- Dividir una columna en varias (por ejemplo, columna “Nombre: García, Miguel”; pasar a dos columnas “Nombre: Miguel” y “Apellido: García”).

- La aplicación de cualquier forma, simple o compleja, de validación de datos y la consiguiente aplicación de la acción que en cada caso se requiera:
 - Datos OK: entregar datos a la siguiente etapa (carga en la aplicación).
 - Datos erróneos: ejecutar políticas de tratamiento de excepciones.

Para la transformación generamos el archivo “TRANS_IBOPE5.qvw” que contiene el script con el código para este proceso y tiene el aspecto como se muestra en la siguiente imagen. Figura.3.18:

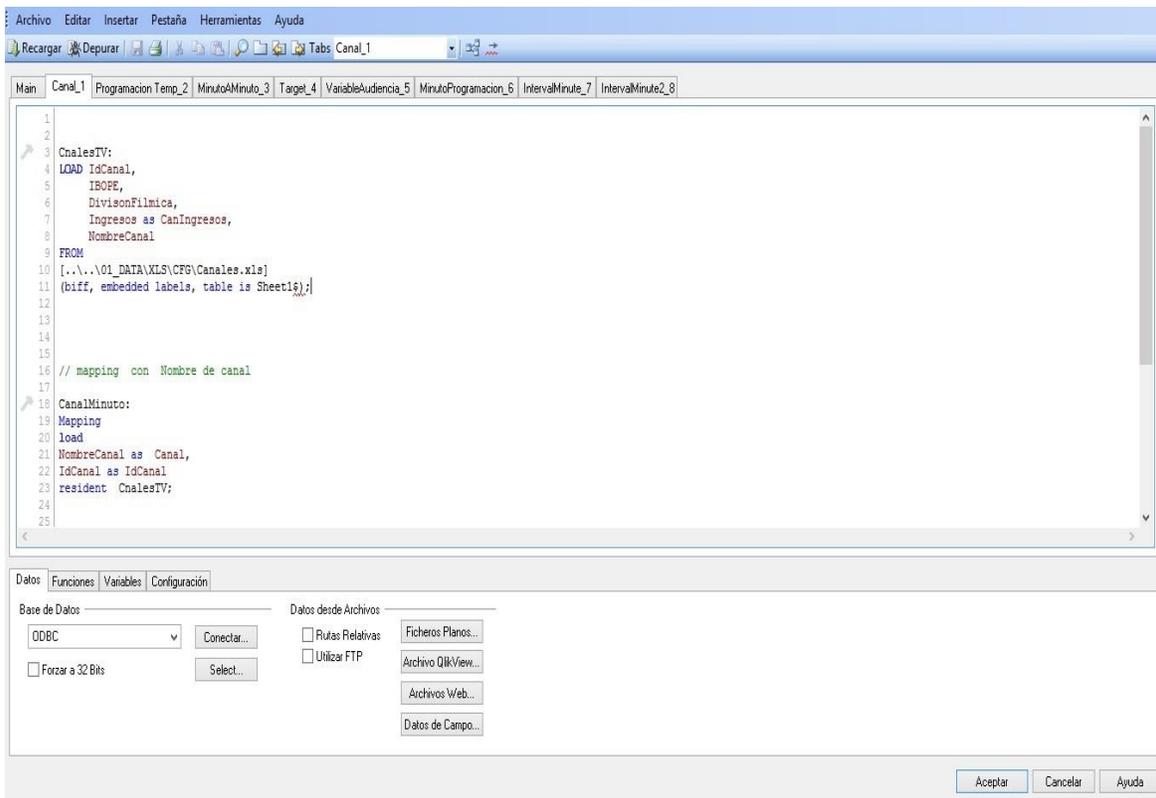


Figura: 3.18. Editor del script de la transformación.

El código generado para el proceso de transformación, se puede consultar en el **ANEXO B** del presente documento.

3.3. Modelo asociativo

En la construcción del sistema nos enfocamos mayormente en la creación de la interfaz de análisis de la aplicación. Sin embargo, una parte fundamental del desarrollo de aplicaciones Qlikview es la construcción de un modelo de datos apropiado que soporte los diferentes análisis requeridos por el usuario.

El corazón de un sistema Qlikview es su modelo de datos que se compone de distintas tablas cuya información puede provenir de diferentes fuentes y se construye usando el lenguaje de programación propio de Qlikview.

Bien diseñado, el modelo de datos asocia todas las tablas de una forma que permite manipular los datos a gusto del usuario. Así la posibilidad de crear objetos de análisis (gráficos) con distintas dimensiones depende de cómo se construya el modelo de datos y de cómo se asocien las respectivas tablas (es decir, cómo se ligen entre sí).

Se podría decir que el modelo asociativo es equivalente en estructura a un modelo dimensional. Sin embargo, un modelo de datos de cualquier tipo en Qlikview se vuelve un modelo asociativo porque no sólo contiene las distintas tablas fuente, de las cuales los gráficos obtendrán los datos, sino que también mantiene estas tablas asociadas permitiendo al documento Qlikview y a sus usuarios explotar su información y realizar agregaciones de datos y cruces de dimensiones para visualizarlos desde muchas perspectivas. En un modelo de datos de Qlikview, todos los valores de campos de todas las tablas en el modelo están asociados entre sí de forma automática en base solamente al nombre de los campos.

Pueden dos o más campos estar almacenados en tablas de dimensión diferentes en el modelo de datos y el hecho de que estén asociadas permite que Qlikview realice la referencia cruzada de forma natural, dando pie al análisis asociativo.

En un modelo de datos asociativo, cualquier campo puede actuar como dimensión en un gráfico. Así mismo, todos los campos pueden también ser usados en expresiones de gráfico para realizar agregaciones de sus valores.

Es importante considerar los siguientes lineamientos para la asociación de tablas:

Qlikview crea asociaciones entre tablas mediante el nombre de los campos y éstos son sensibles a mayúsculas y minúsculas.

Qlikview construye el modelo de datos con sus respectivas asociaciones, incluso si el desarrollador no define explícitamente cómo desea que las tablas se ligen. Sin embargo, para que esto suceda, las tablas fuente deben tener los nombres de campo correctos. Como este escenario lo encontramos en raras ocasiones, especialmente si estamos cargando tablas desde diferentes sistemas fuente, la herramienta más básica y fundamental para el diseño de un modelo de datos en Qlikview es renombrar campos.

Renombre de campos:

Hay dos escenarios principales para renombrar un campo:

- Asegurar que dos tablas se asocien a través de los campos correctos cuando dichas tablas no tienen un campo en común, es decir, con el mismo nombre, pero a pesar de ello es posible asociar estas dos tablas mediante algún par de campos.
- Prevenir asociaciones no deseadas entre tablas cuando éstas comparten un campo con el mismo nombre, pero dicho campo no representa realmente una llave con la cual se pueda asociar el contenido de las tablas.

Para renombrar un campo, simplemente se utiliza la palabra reservada “as” en el script de carga para asignar un alias al nombre de campo original. O podemos utilizar la instrucción “Qualify” que puede ser usada para calificar nombres de campo con el nombre de su tabla correspondiente, lo cual significa que los campos especificados serán renombrados para darles la forma de “nombredetabla.nombredcampo”, asegurando así que no se creen asociaciones no deseadas.

En algunos casos, podemos requerir renombrar un número mayor de campos y resulta cada vez más impráctico el listarlos todos como parte de la instrucción. Afortunadamente, la instrucción “Qualify” también permite el uso de caracteres comodín en el parámetro de “fieldlist”. Por ejemplo, podemos usar un asterisco para especificar que todos los campos cargados a partir de la instrucción y en adelante sean calificados. Podemos también combinar el asterisco con una cadena o con un signo de interrogación (el cual es otro carácter comodín) para especificar que un conjunto de campos cuyos nombres coincidan con un criterio dado sean calificados. Por ejemplo:

- “Qualify_Código, Descripción;”: Esta instrucción calificará solamente los campos de nombreCódigo y Descripción.
- “Qualify “*ID”;”: Este comando calificará todos los campos cuyo nombre termina con ID.
- “Qualify *;”: Este comando calificará todos los campos que sean cargados desde ese punto en adelante.

Algunas veces, la función “Qualify” requiere ser activada solo para una parte del script, pero se debe desactivar después de cargar ciertas tablas. Para hacer eso, podemos simplemente usar la instrucción “Unqualify”, especificando la lista de campos de la misma manera como se describió anteriormente.

En otros casos, requeriremos que la función de “Qualify” aplique para todos los campos excepto uno o dos (por ejemplo, los campos llave de forma que se mantengan las asociaciones). Para lograr esto, podemos combinar las instrucciones Qualify y Unqualify como se hace en el siguiente ejemplo:

```
Qualify *;
```

```
Unqualify %Campo_Llave1, %Campo_Llave2;
```

La combinación de instrucciones mostrada arriba provocará que todos los campos cargados desde ese punto en adelante sean calificados con sus respectivos nombres de tabla, excepto los campos llamados “%Campo_Llave1” y “%Campo_Llave2”. Esto es muy útil cuando queremos asegurar que los campos llave no se vean afectados por la instrucción “Qualify”.

La instrucción “Qualify” puede ser desactivada en cualquier punto del script con una instrucción “Unqualify”.

Dos problemas frecuentes en la construcción del modelo de datos asociativo son:

- A) La presencia de lo que conocemos como "llaves sintéticas" (descritas más adelante).
- B) La presencia de referencias circulares en el modelo de datos.

Éstos pueden causar una degradación en el desempeño de la aplicación Qlikview o incluso inconsistencia de datos.

Llaves sintéticas

Cuando dos tablas comparten más de un campo en común, Qlikview crea una llave compuesta, o llave sintética, para tratar de asociar ambas tablas a través de la combinación de valores de todos los campos que tienen en común. Esto forma una tabla adicional en el modelo de datos que contiene los campos en común entre las tablas involucradas, así como una llave compuesta de dichos campos que también es añadida a las tablas correspondientes para crear la asociación.

La presencia de llaves sintéticas en un modelo de datos puede causar que la aplicación tenga tiempos de respuesta altos y algunas veces incluso que se consuman todos los recursos disponibles. Por lo tanto, se sugiere evitarlas cuando sea posible.

Algunos métodos que se pueden utilizar para remover llaves sintéticas de un modelo de datos:

- Renombrar los campos que forman la llave sintética y que no deberían formar parte de la asociación entre las dos tablas.
- Remover los campos que causan la presencia de la llave sintética de una de las dos tablas. Para remover un campo, simplemente se borra la línea de código correspondiente del script de carga.
- Crear una llave compuesta, representada por un nuevo campo calculado en base a la concatenación de todos los campos comunes que realmente se requieren para ligar ambas tablas.
- Después de crear una nueva llave compuesta, podemos borrar los campos individuales de cualquiera de las dos tablas.

El siguiente diagrama de flujo, véase Figura 3.19, muestra el proceso que un desarrollador debe seguir para decidir cuál de los métodos mencionados arriba se debe utilizar:

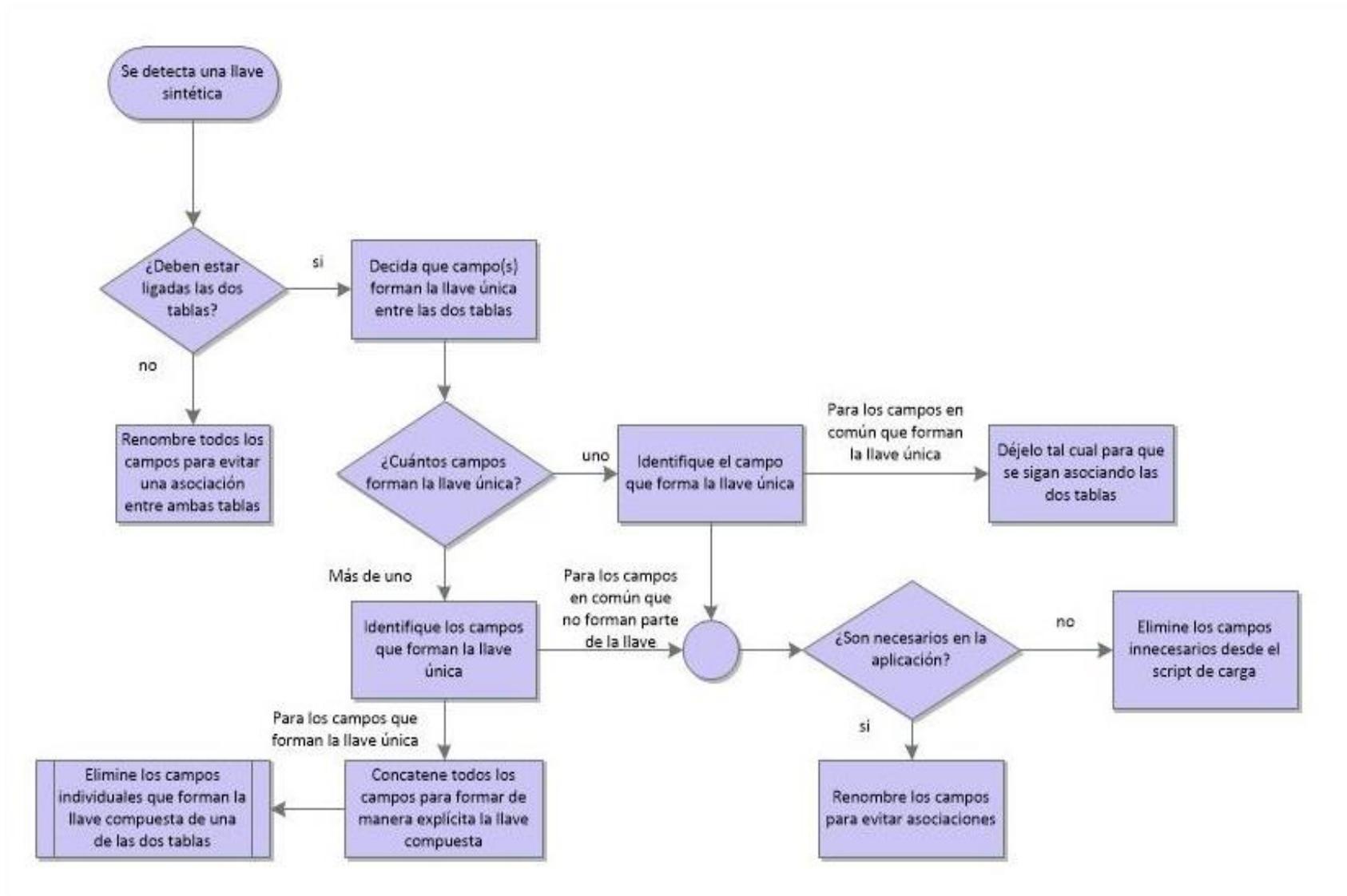


Figura3.19. Diagrama de flujo de un proceso de decisión para asociación de tablas.

La llave compuesta se crea desde el script de carga. Ver figura 3.20.

```

// create key
Programacion:
load
( upper(Dominio) & IdCanal & Date & Target & Variable & PrgTiempoInicio & PrgTiempoFin) as keyProgramacion,
( upper(Dominio) & IdCanal & Date & Target & Variable ) as keyPrograMin,
( upper(Dominio) & IdCanal & Date & Target & Variable ) as keyPrograMin2,
*
resident ProgramacionTemp2;

drop table ProgramacionTemp2;

//exit Script;
    
```

Figura 3.20. Script de carga.

Al concatenar campos, siempre es una buena idea usar un separador entre ellos para asegurar la consistencia de datos.

La figura 3.21 muestra la llave compuesta generada en el modelo de datos asociativo del sistema:

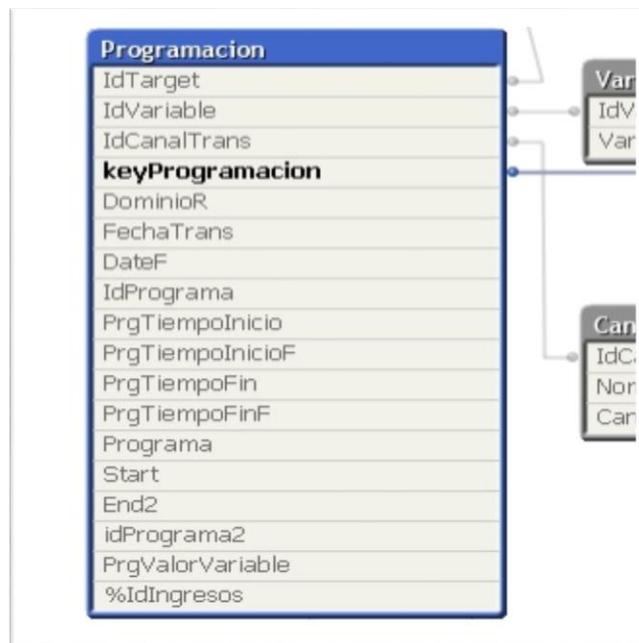


Figura 3.21. Llave compuesta del modelo asociativo.

Referencias circulares

Una referencia circular también puede ser resultado de una asociación de tablas inadecuada en nuestro modelo de datos y como tal, puede ser corregida usando los mismos principios descritos previamente.

Podemos ver a nuestro modelo de datos como un mapa que muestra los caminos a través de los cuales podemos llegar de un punto a otro, debemos asegurar que siempre haya una única ruta entre dos puntos cualesquiera. De otro modo, estaríamos lidiando con una referencia circular.

Una referencia circular puede crear problemas de consistencia de datos, degradación de desempeño de la aplicación y en algunos casos hasta hacer inoperable la aplicación.

Finalmente el modelo asociativo de nuestro sistema quedó como se muestra en la figura 3.22.

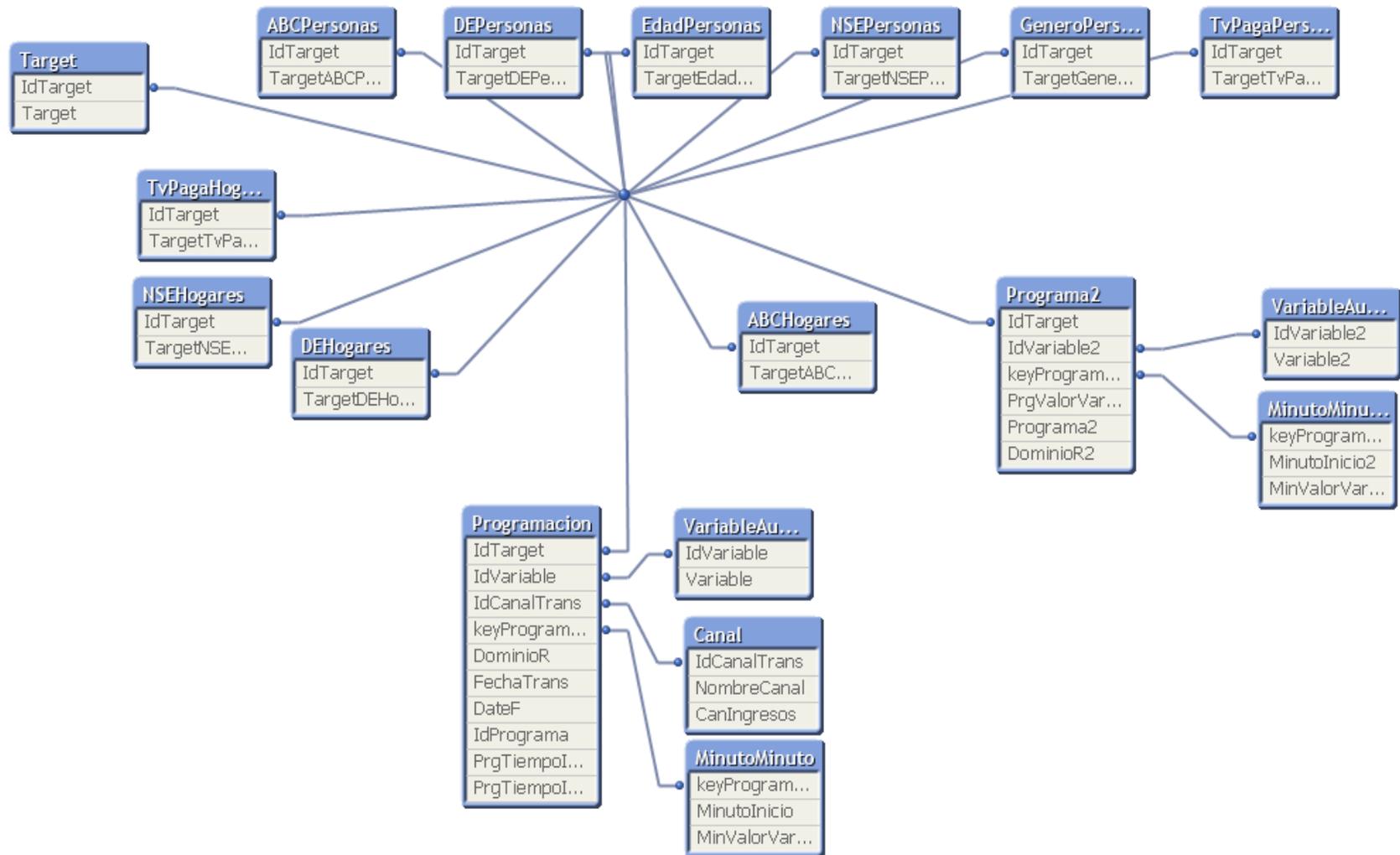


Figura 3.22. Modelo asociativo de nuestro sistema.

3.4. Desarrollo de la aplicación

Considerando los requerimientos funcionales definidos en el capítulo II de este documento y después de haber llevado a cabo los procesos de extracción y transformación de datos, lo que nos llevó a nuestro modelo asociativo, creamos el archivo “SISTESIS.qvw” que será el archivo ejecutable de nuestro sistema y que nos permitirá crear los objetos de la siguiente manera:

PANTALLA 1: BIENVENIDA

Descripción: al ejecutar el documento “SISTESIS.qvw” y estando en el área de trabajo comenzamos con la configuración de nuestra pestaña que tendrá como título “BIENVENIDO” como se muestra en la siguiente figura 3.23.

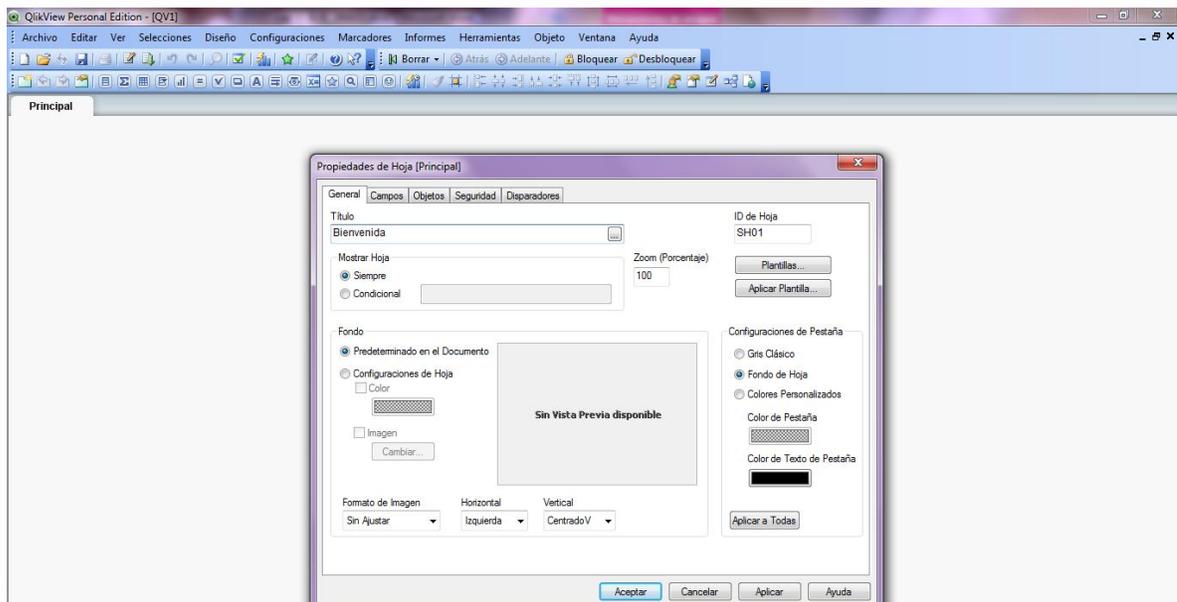


Figura 3.23. Configuración de la primera pestaña del sistema.

En seguida comenzamos con la construcción de los objetos como se muestra en la figura 3.24, la figura 3.25. y figura 3.26.

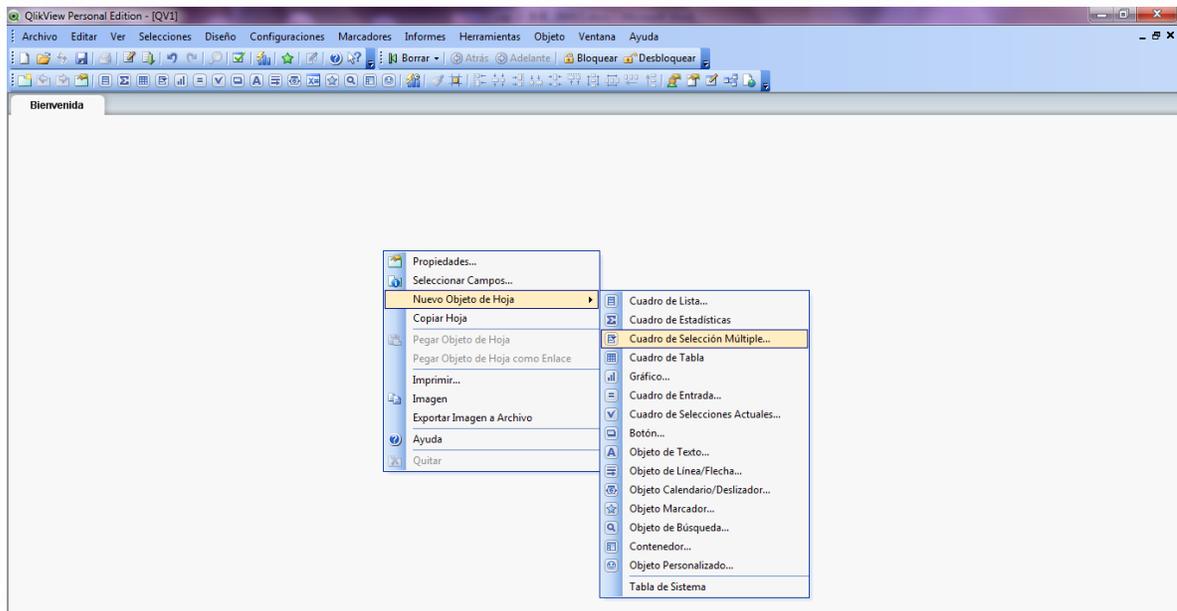


Figura 3.24. Selección del tipo de objeto.

Lo que se muestra en la figura 3.24 es el resultado de dar click con el botón derecho del mouse que es el listado de objetos que pueden seleccionarse en la construcción de nuestro sistema.

Para colocar el escudo de la institución, seleccionamos el tipo “Objeto de texto”, activamos la casilla “Imagen” y buscamos nuestra imagen en nuestro equipo a través del botón “Cambiar” como se muestra en la figura 3.25.

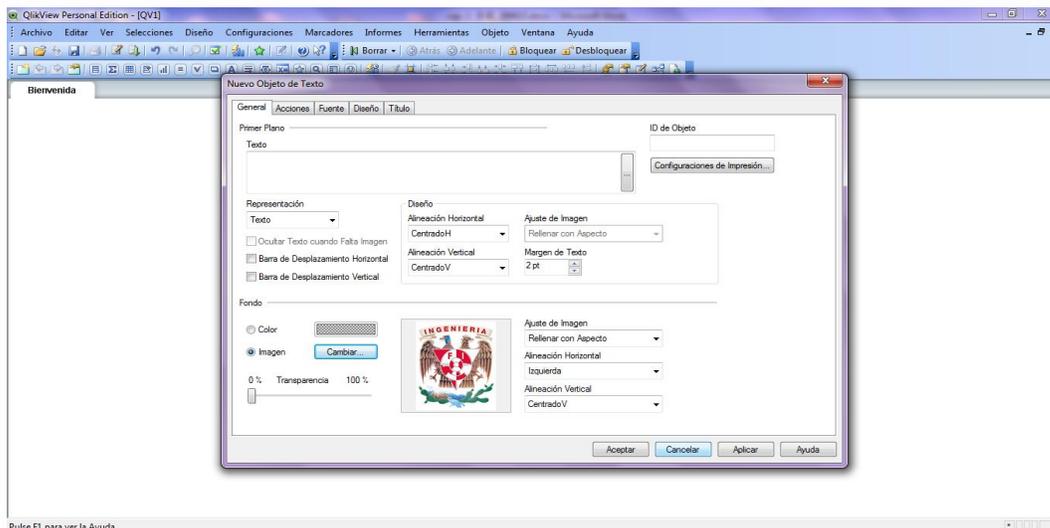


Figura.3.25. Objeto de texto para la imagen.

Para el título del sistema, también seleccionamos un “Objeto de texto”, colocamos la información requerida y dentro de sus propiedades es posible establecer el formato del texto como se muestra en la figura 3.26.

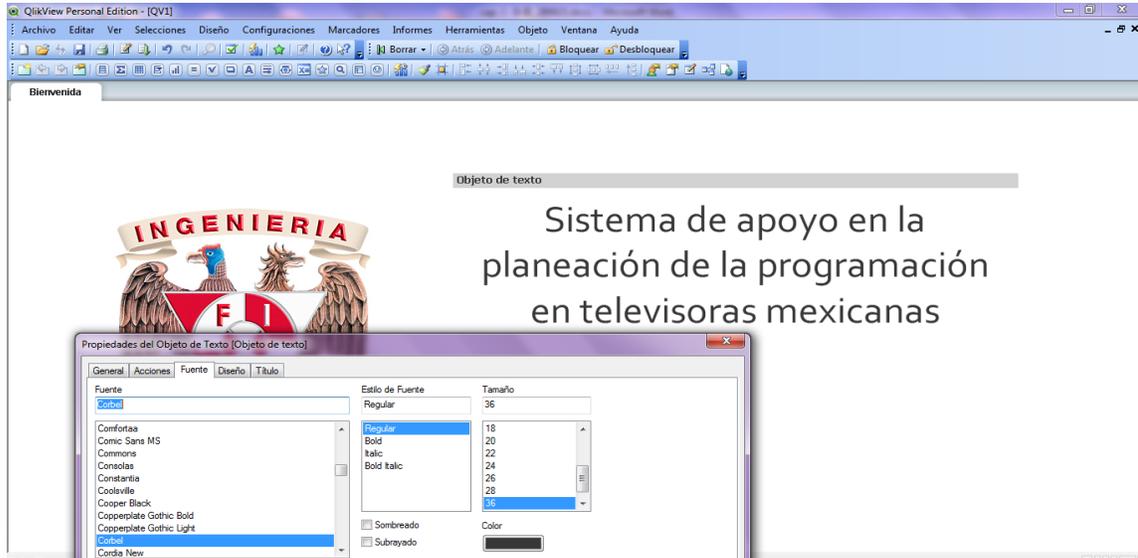


Figura 3.26. Objeto de texto para el título del sistema.

Finalmente nuestra pestaña “BIENVENIDO” queda como se observa en la figura 3.27.

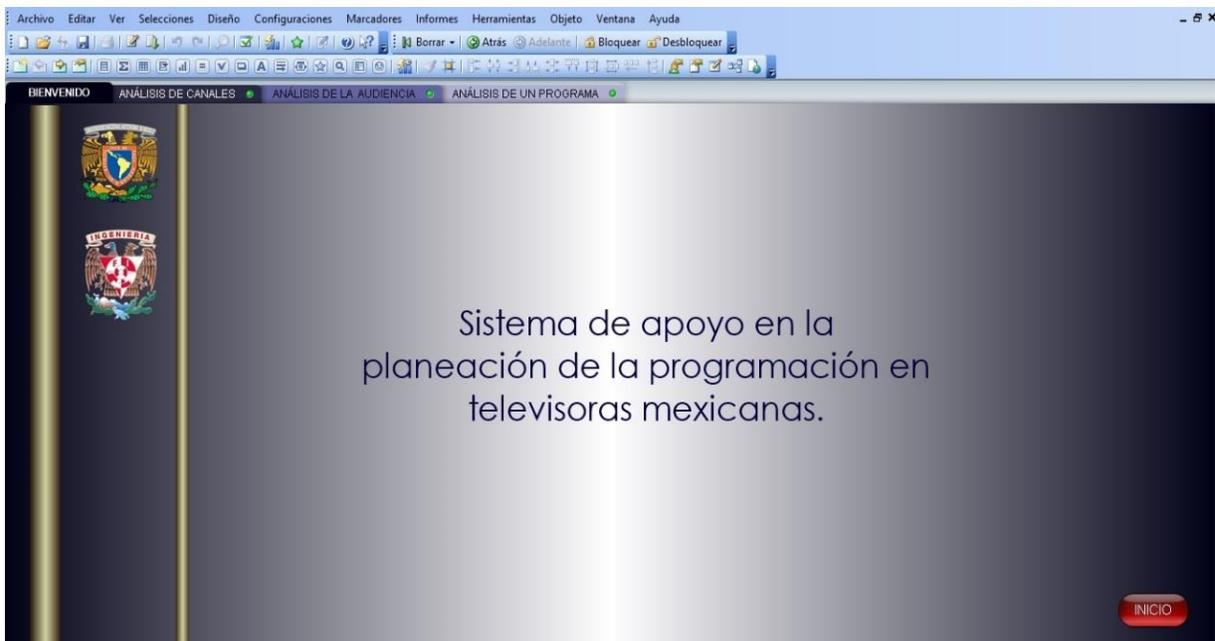


Figura 3.27. Pestaña “BIENVENIDO” de nuestro sistema.

PANTALLA 2: ANÁLISIS DE CANALES

Descripción: Para generar la segunda pestaña “ANÁLISIS DE CANALES” añadimos una nueva hoja como muestra la figura 3.28.

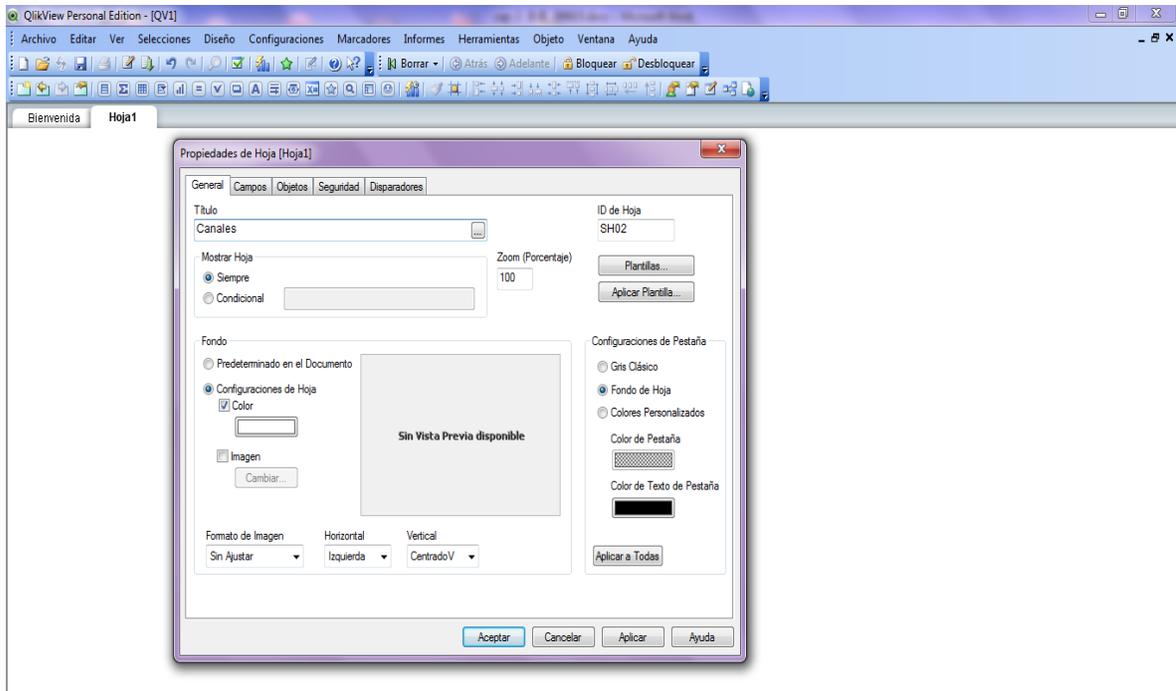


Figura 3.28. Creación de la segunda pestaña.

Enseguida creamos los objetos que servirán para seleccionar los parámetros que mostrarán la información en los gráficos “A” y “B”, para ello seleccionamos “Cuadros de lista” que muestran la información de los campos: “REGIÓN”, “TARGET”, “VARIABLE” y “FECHA”, después creamos los objetos gráficos como lo muestra la figura 3.29.

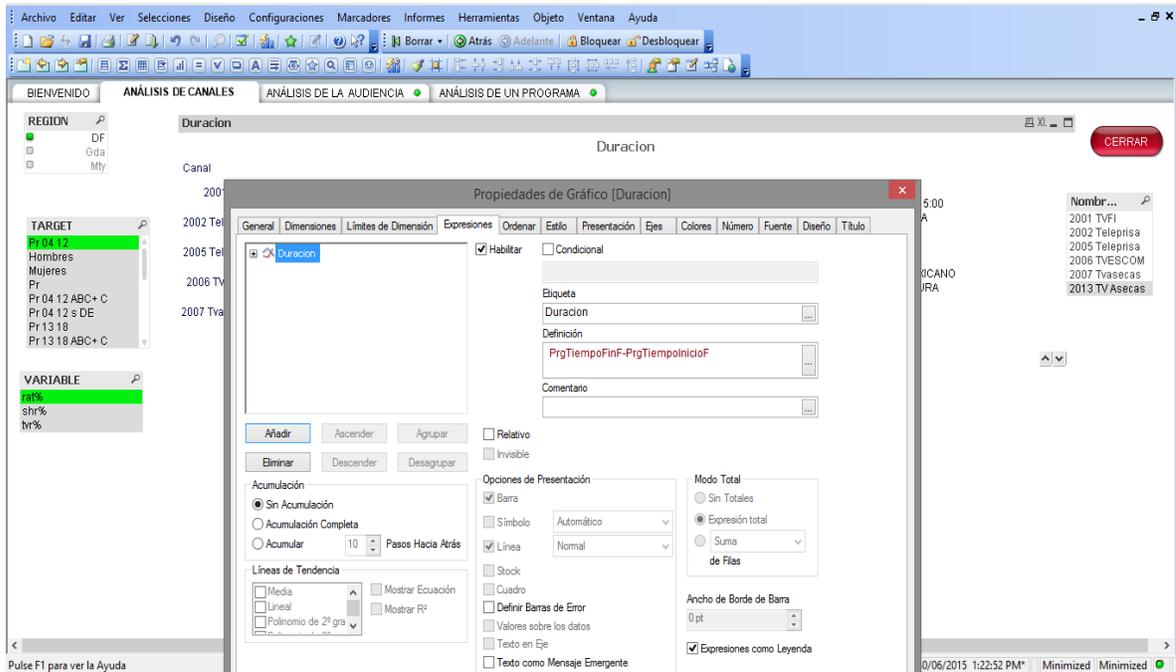


Figura 3.29. Creación de cuadros de lista y gráficos en segunda pestaña.

Quedando al final la pestaña “ANÁLISIS DE CANALES” como se observa en la figura 3.30.

PANTALLA 3: ANÁLISIS DE LA AUDIENCIA

Descripción: Para generar la tercera pestaña “ANÁLISIS DE LA AUDIENCIA” añadimos una nueva hoja como muestra la figura 3.31.

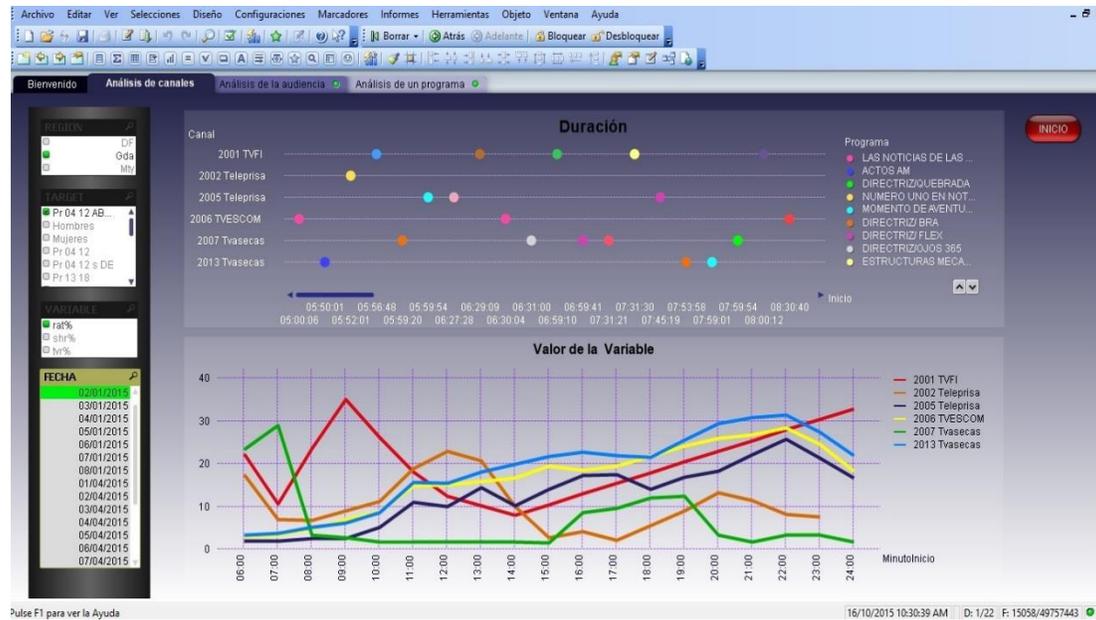


Figura 3.30. Pestaña “ANÁLISIS DE CANALES”.

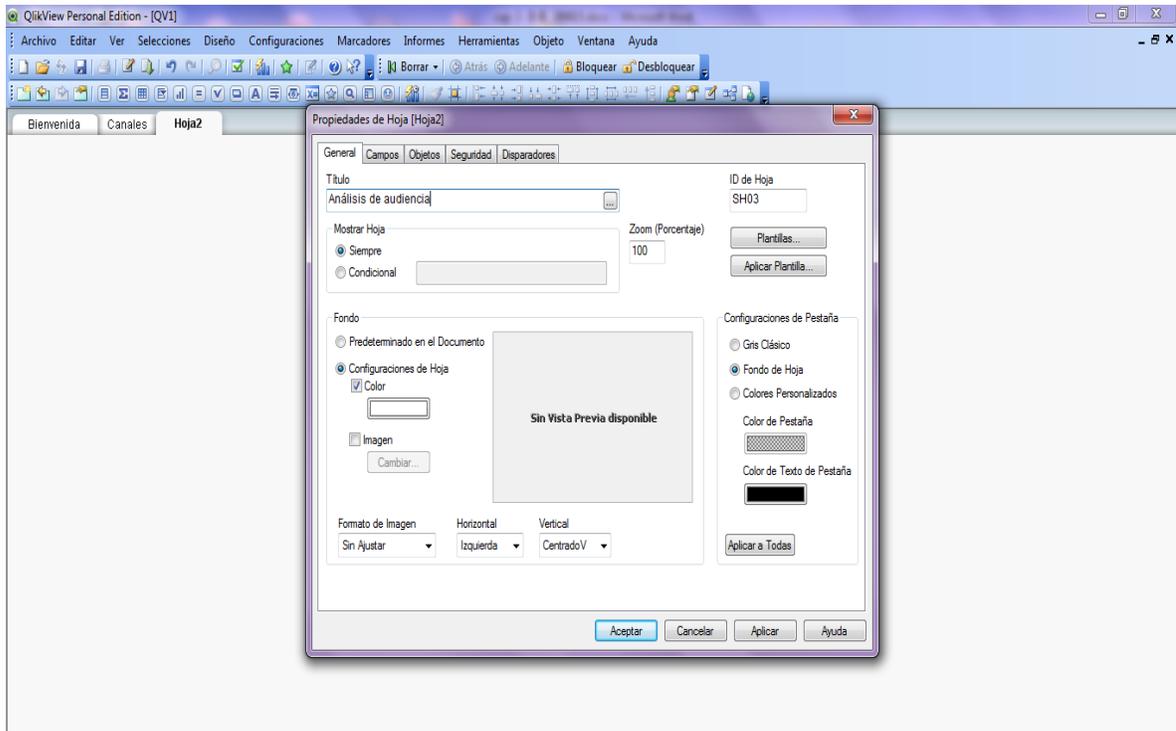


Figura 3.31. Creación de la pestaña “ANÁLISIS DE LA AUDIENCIA”.

Al igual que en el proceso de creación de la segunda pestaña, se seleccionan objetos “Cuadros de lista” sólo que para esta pestaña la información mostrada será la de los campos: “REGIÓN”, “PROGRAMA”, “VARIABLE1” y “VARIABLE2”.

La información se presentará de acuerdo a los perfiles: “Género”, “Edad”, “Nivel Socio-económico” y “Con o sin TV de paga” por lo que se es necesario crear objetos de tipo “Gráfico” con sus respectivas configuraciones, como se muestra en la figura 3.32.

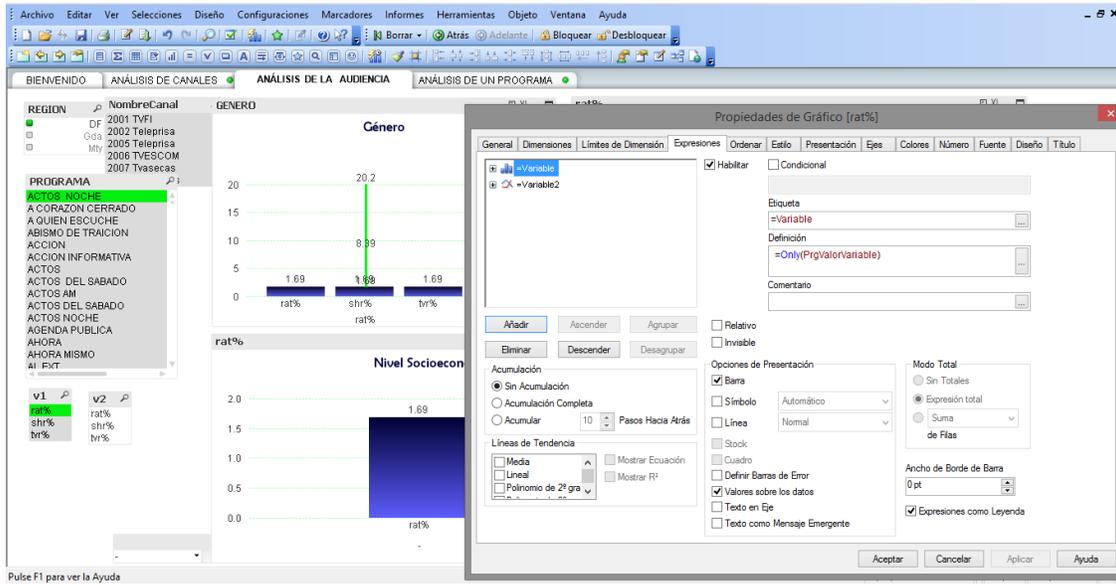


Figura 3.32. Configuración de gráficos de la tercera pestaña.

Finalmente la pestaña “ANÁLISIS DE LA AUDIENCIA” queda como se observa en la siguiente figura.3.33.

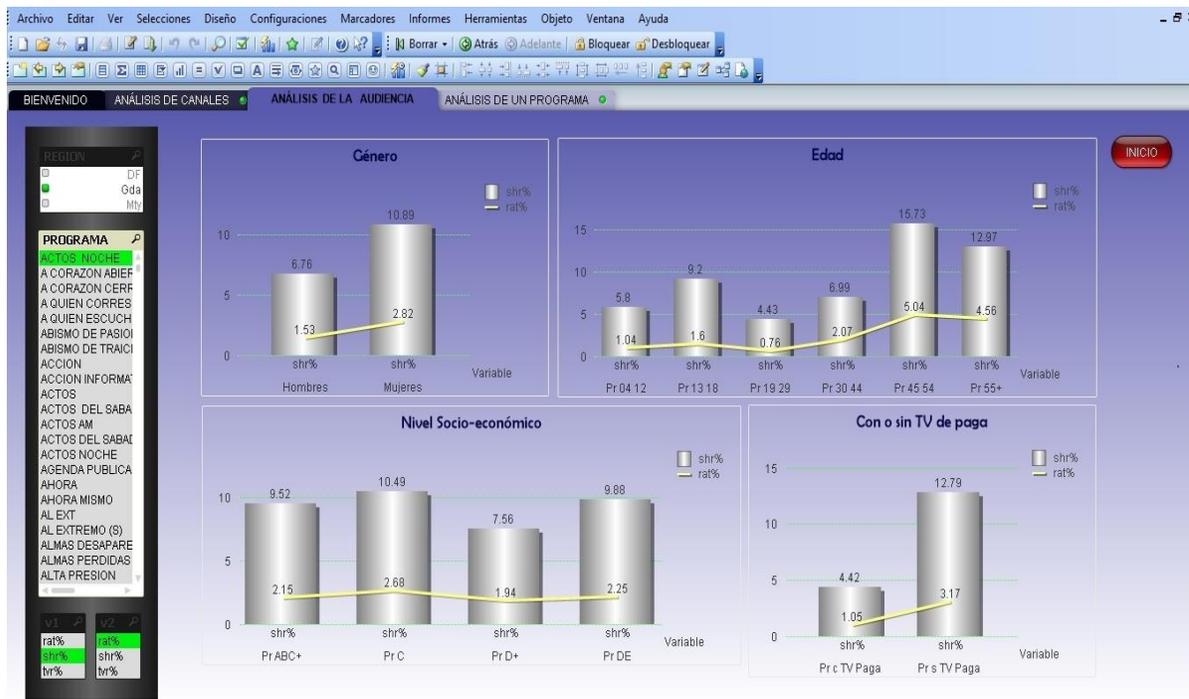


Figura 3.33. Aspecto final de la pestaña “ANÁLISIS DE LA AUDIENCIA”.

PANTALLA 4: ANÁLISIS DE UN PROGRAMA

Descripción: para generar la cuarta pestaña “ANÁLISIS DE UN PROGRAMA” añadimos una nueva hoja como muestra la figura 3.34.

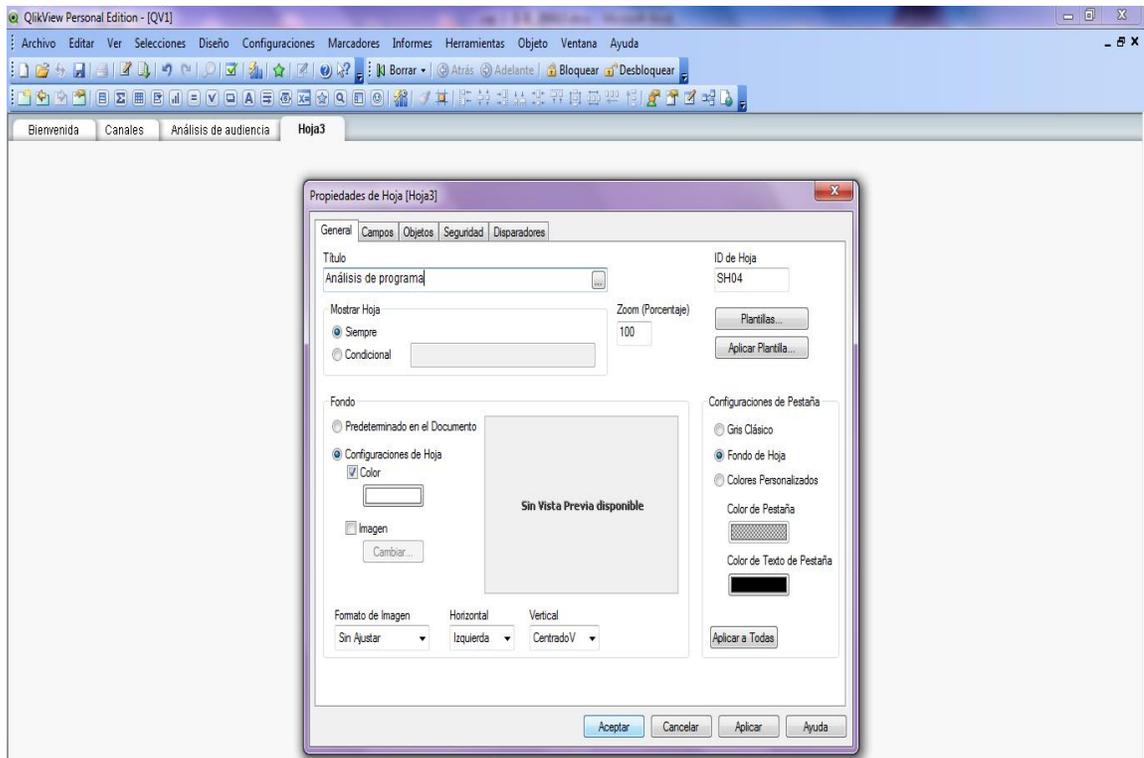


Figura 3.34. Creación de la pestaña “ANÁLISIS DE UN PROGRAMA”.

En seguida creamos los objetos como lo hicimos en la generación de las pestañas anteriores. Los campos a utilizar en esta cuarta pestaña son: “REGIÓN”, “VARIABLE”, “PROGRAMA” y “TARGET”. De igual forma se crean objetos de tipo “Gráfico”, los configuramos para finalmente mostrarlos como se observa en la figura 3.35.

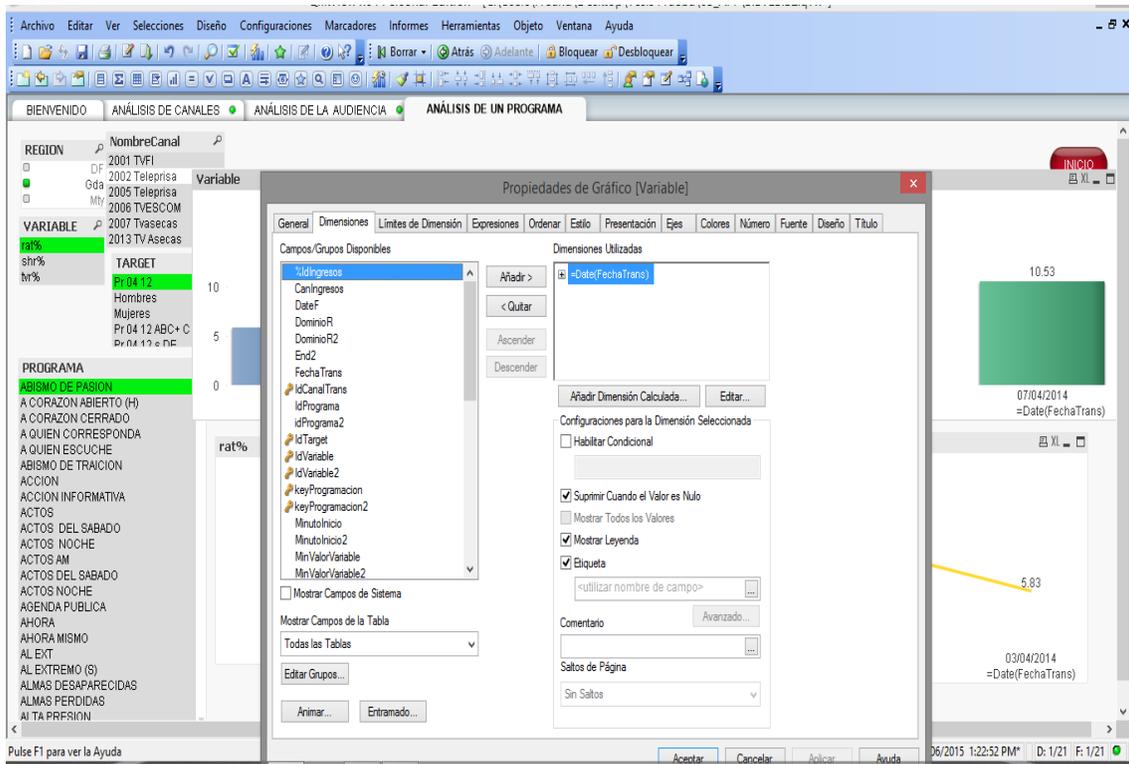


Figura 3.35. Creación de objetos en la cuarta pestaña.

Finalmente la pestaña “ANÁLISIS DE UN PROGRAMA” queda como se observa en la figura 3.36.

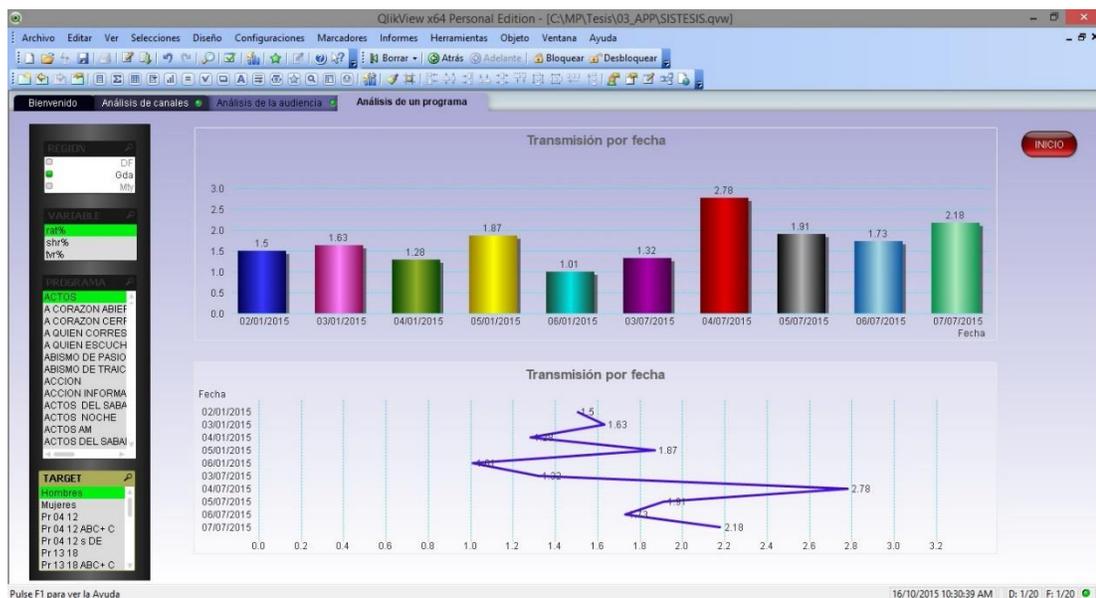


Figura 3.36. Aspecto final de la pestaña “ANÁLISIS DE UN PROGRAMA”.

Para fines demostrativos, el sistema contendrá información histórica de 3 semanas, elegidas de la siguiente manera:

Semana del 01 al 07 del mes de enero de 2015

Semana del 01 al 07 del mes de abril de 2015 y,

Semana del 01 al 07 del mes de julio de 2015.

Nota: todos los datos contenidos en los archivos fuente no son reales.

3.5. Validación de datos y funcionalidad.

El mecanismo de validación de datos consiste en verificar que el valor de los datos contenidos en los archivos fuente sea el mismo que se muestra en el sistema después de haber aplicado los filtros correspondientes.

Para ello elegimos 8 valores al azar, cuyos filtros y resultados se muestran en la siguiente tabla 3.1.

Ítem	Fecha	Pestaña	Región	Variable	Target	Hora	Programa/ Canal	Valor Sistema	Valor fuente
1	01/04/2015	Análisis de programa	DF	Rat%	Hombres	NA	Actos AM	1.46	1.46
2	03/01/2015	Análisis de programa	GDA	Tvr%	Pr DE	NA	Lo que ocultan las mujeres	19.24	19.24
3	02/01/2015	Análisis de audiencia	MTY	Shr%	Pr C TV Paga	NA	Noticieros Pérez	6.38	6.38
4	03/01/2015	Análisis de programa	DF	Tvr%	Pr S TV Paga	NA	Huerfanitas	2.71	2.71
5	03/01/2015	Análisis de audiencia	GDA	Shr%	Pr DE	NA	Venga las sonrisas	12.05	12.05
6	01/07/2015	Análisis de audiencia	MTY	Shr%	Pr ABC+	NA	Traje Rojo	7.17	7.17
7	03/01/2015	Canales	DF	Rat%	Pr C	09:00	TVFI	0.53	0.53
8	01/07/2015	Canales	MTY	Tvr%	Pr 55+	17:00	2002 Teleprisa	35.43	35.43

Tabla 3.1. Tabla de validación de datos.

La validación de la funcionalidad consistió en la realización de mediciones de los tiempos de apertura y ejecución del sistema, tomando como muestra cuatro de sus objetos que lo conforman. Se observó que estos se carguen correctamente y en el menor tiempo posible.

En la tabla 3.2 se presentan los resultados de las mediciones obtenidas.

Id de objeto	Nombre	Pestaña	Tiempo de carga	Tiempo de ejecución
NA	Síntesis	NA	17s	3s
CH01	Duración	Análisis de canales	3s	1s
CH11	Género	Análisis de audiencia	4s	2s
CH13	Con/Sin TV Paga	Análisis de audiencia	4s	2s
CH14	Transmisión por fecha	Análisis de programa	3s	2s

Tabla 3.2. Tabla con resultados de las mediciones obtenidas.

Capítulo IV

IMPLANTACIÓN Y CIERRE

4.1. Configuración de ambiente productivo

El software utilizado (Qlikview) trabaja con esquema de licenciamiento y es posible utilizar de forma personal y gratuita la versión descargable desde el sitio oficial de la empresa QlikTech (www.qlikview.com).

En esta tesis utilizamos la versión 11 instalada en un dispositivo personal que fungió como servidor de prueba, el cual tenía las especificaciones técnicas mostradas en la tabla 4.1.

Servidor de prueba	
Procesador	Core i5
Sistema Operativo	Windows 8.1
Memoria	Ram 4 GB
Disco Duro	300 GB

Tabla 4.1. Especificaciones técnicas del servidor de prueba.

Una vez concluido el sistema y para poder ser utilizado, este debe instalarse en un servidor de producción “dedicado” que debe contar con al menos las siguientes características:

- Procesador Pentium II ó más alto.
[QlikView hace el pleno uso de arquitectura del procesador Moderno multi-Core.]
- Hardware de gráficos que soporta resolución XGA.
- Un monitor con una resolución de al menos 1024 x 768.
- Un ratón o un dispositivo equivalente soportado por Windows.
- Un lector de CD-ROM (si la instalación debe ser hecha del CD).
- Disco duro con al menos 1 TB de espacio de disco libre.
- Memoria RAM: 4 GB como mínimo.

Así mismo debe instalarse una versión para servidor de Qlikview server, para ello se necesitan los siguientes pre-requisitos:

- 1) Acceso con un usuario con derechos de administrador local
- 2) Framework .NET v3.5.x SP1 (no aplica otra versión y además debe contar con el 3.5 SP1)
- 3) Windows 2003 Server ó 2008 x64, IIS 6 ó 7 (Si es 7 se necesitan algunos componentes particulares) o se puede utilizar el Qlikview Web Server
- 4) Usuario de dominio para los servicios con un password que no expire que además sea administrador local del equipo. (Puede ser usuario local pero conviene que sea de dominio para que pueda acceder a carpetas de red).
- 5) Habilitar puerto 4780 para acceso remoto a la consola de administración server/Publisher, puertos 80 y 4747 para acceso a las aplicaciones.
- 6) Cliente Oracle u otra base de datos que se tenga que acceder.
- 7) IE 7, Firefox 3.0 ó Chrome.
- 8) Modo UAC deshabilitado en Windows 2008.
- 9) IP Fija y nombre de red definitivo.
- 10) Servicio "Print Spooler" habilitado (o no se instalará correctamente la impresora a pdf).
- 11) Obtener autorización para crear el grupo local "Qlikview Administrators". Obtener autorización para agregar al usuario del servicio como administrador local.

La instalación de la versión del servidor (Qlikview server) debe realizarse de la siguiente manera:

Hacer doble click en el instalador y aparecerá una pantalla de extracción del archivo .MSI como se muestra en la figura 4.1.



Figura 4.1. Instalación de Qlikview server.

Enseguida aparecerá el wizard de instalación y dar click en el botón "Siguiente", como se muestra en la figura 4.2.



Figura 4.2. Pantalla de wizard de Qlikview.

Leer el contrato de licencia y aceptar los términos de licencia. Si fuera necesario, imprimir el contrato de licencia y dar click en “Siguiente”. Ver figura 4.3

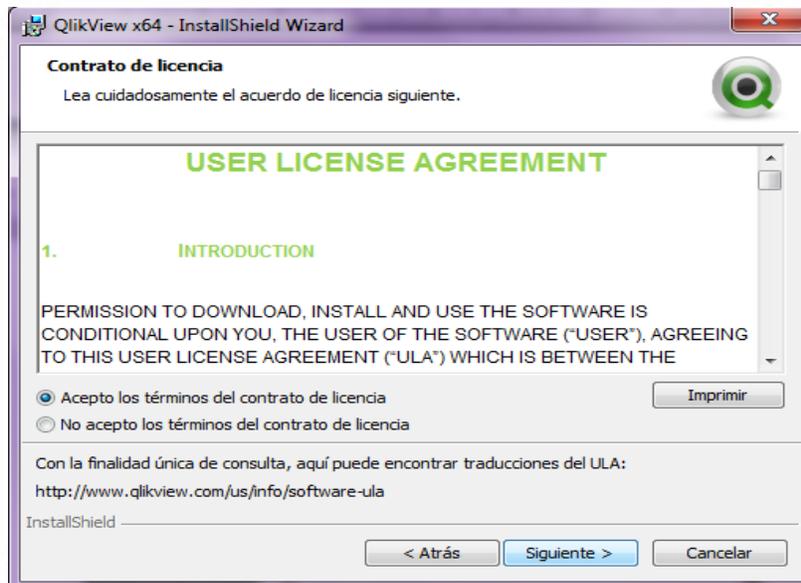


Figura 4.3. Pantalla de aceptación de licencia de Qlikview.

En la figura 4.4 se muestra la imagen en donde se ingresa el “Nombre de usuario” y “Organización”. Dar click en el botón “Siguiente” para continuar con la instalación.

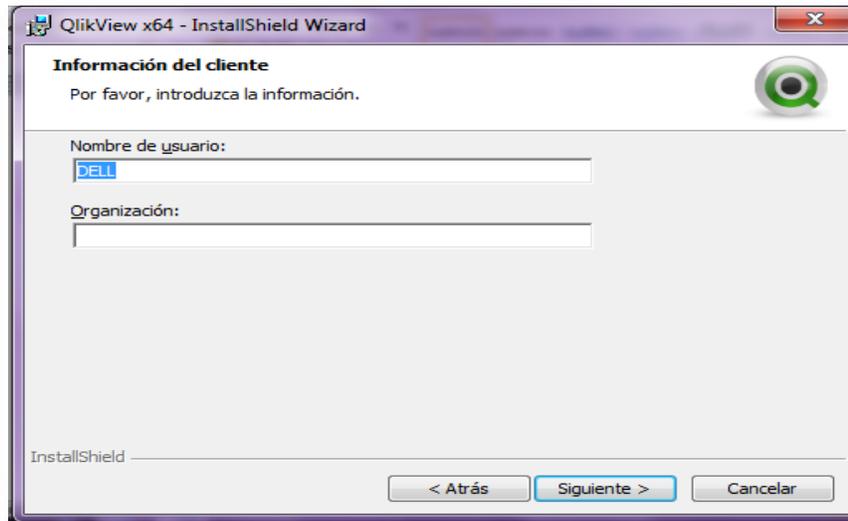


Figura 4.4. Pantalla de usuario y organización de Qlikview.

Si se desea instalar Qlikview Server en una carpeta diferente, es necesario cambiar la dirección de la carpeta destino. La figura 4.5 muestra la imagen en donde aparece la opción indicada. Dar click en “Siguiente” para continuar.

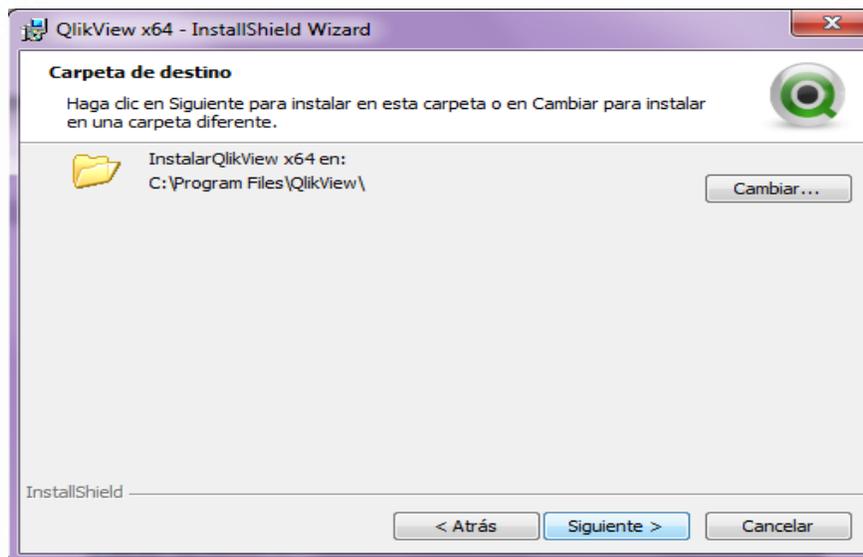


Figura 4.5. Pantalla de ubicación de instalación de Qlikview.

El wizard permite escoger los productos a instalar. Elegir “Completa” o “Personalizada” según sea el caso, como se muestra en la figura 4.6.

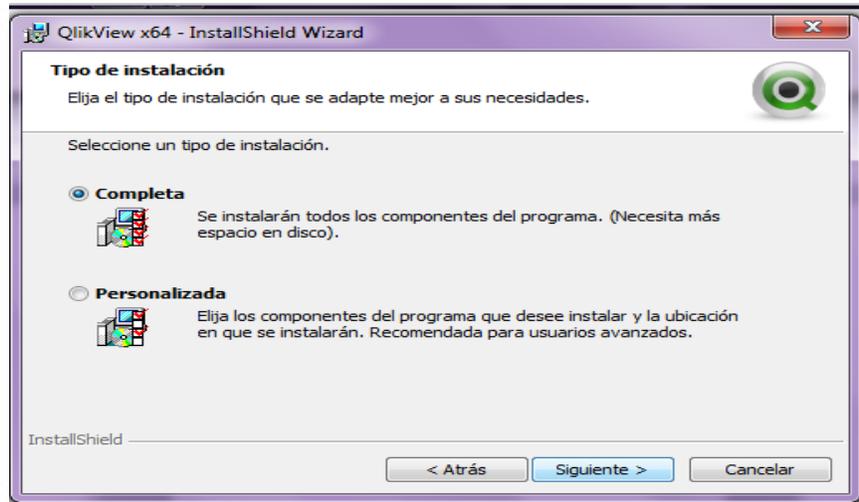


Figura 4.6. Pantalla de selección de programas a instalar en Qlikview.

Es posible elegir el “Soporte de Qlikview” versión única o “Servidor Qlikview” y “Editor”. Si se instala un servidor web se puede escoger entre IIS y el servidor web de Qlikview. El botón configuración otorga la posibilidad de comprobar las opciones a mano. Si se requiere una instalación completa solo es necesario dar click en “Siguiete” o “Next”. Ver figura 4.7.

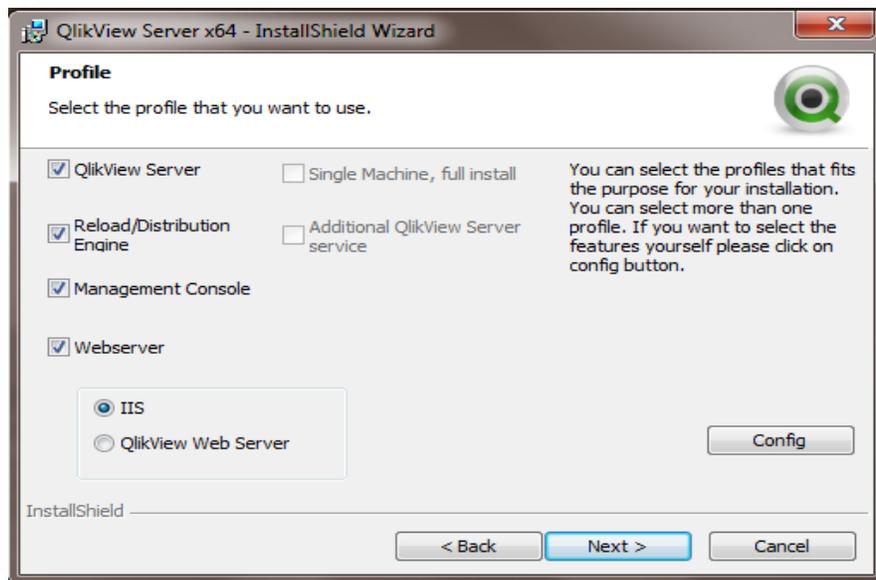


Figura 4.7. Pantalla de selección de servidores en Qlikview.

Si se decide instalar al servidor de web sobre IIS es necesario comprobar las exigencias:

- ASP.NET (Including dependencies)
- Windows Authentication

Seleccionar el sitio web y hacer click en “Next”. Ver figura 4.8.

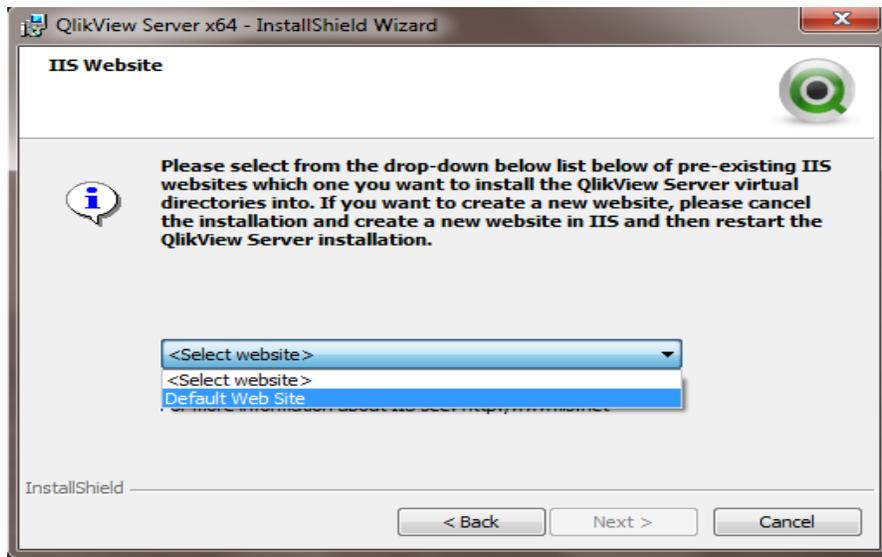


Figura 4.8. Pantalla de selección de servidor web en Qlikview.

A partir de aquí, la instalación es igual para cualquier de las opciones que se haya seleccionado.

Qlikview server necesita una cuenta especial la cual usará y poseerá la aplicación. Esta cuenta será parte de un controlador de Dominio. Hacer click en “Browser”. Ver figura 4.9.

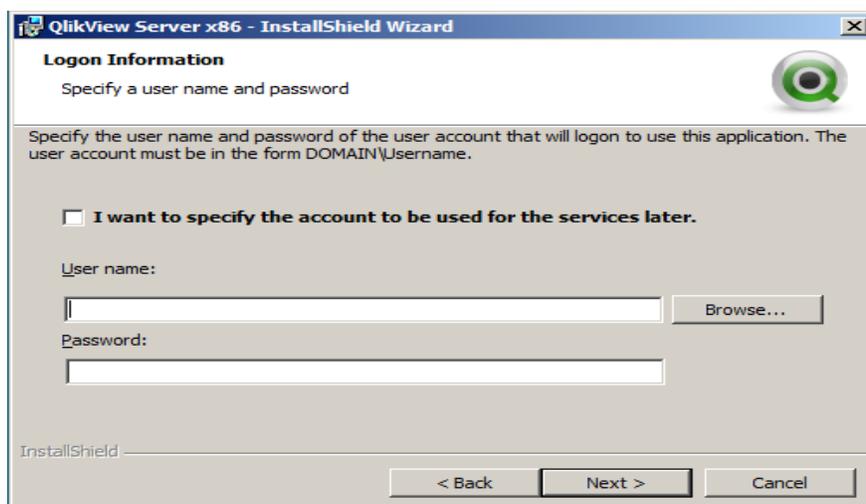


Figura 4.9. Pantalla de ingreso a Qlikview.

Hacer click en los botones “Browse” para seleccionar Dominio y usuario. Y hacer click en “OK”. Ver figura 4.10.

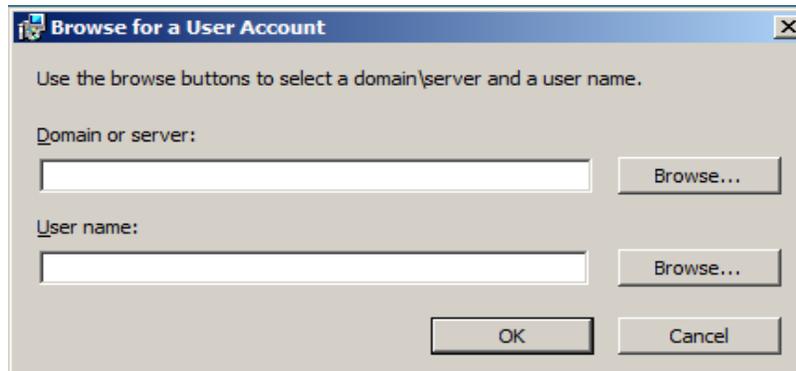


Figura 4.10. Pantalla de selección de dominio en Qlikview.

Ingresa el password del usuario seleccionado y hacer click en “Next”. Ver figura 4.11

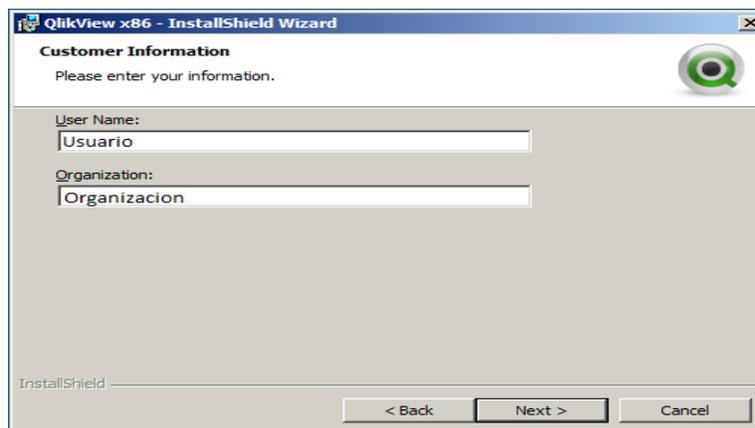


Figura 4.11. Pantalla para ingresar password en Qlikview.

Hacer click en “Instalar” para dar comienzo a la instalación. Ver figura 4.12.

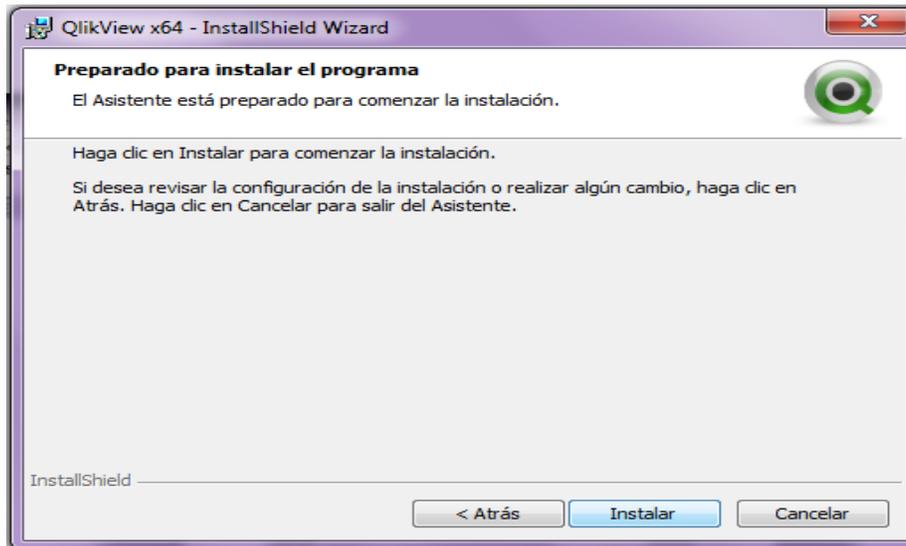


Figura 4.12. Pantalla de instalación de Qlikview.

Qlikview 11 Server está siendo instalado. Se puede cancelar la Instalación de ser necesario. Ver figura 4.13.

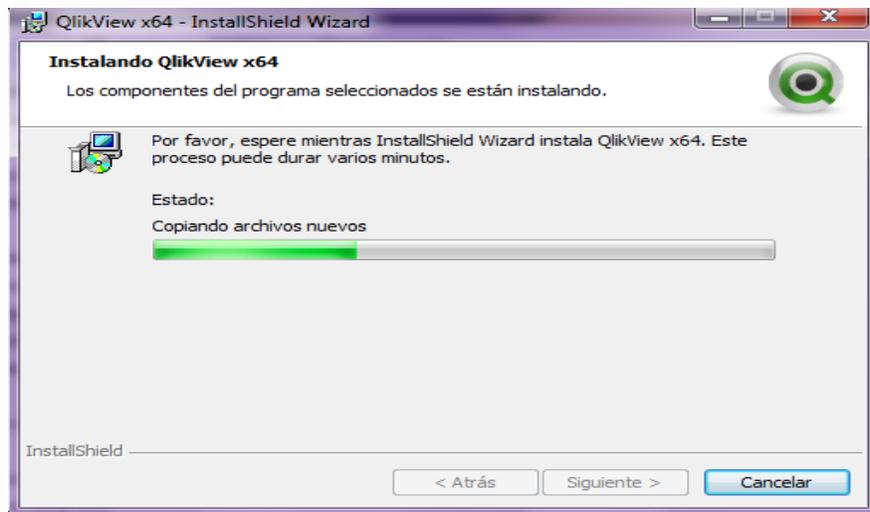


Figura 4.13. Pantalla de seguimiento de instalación de Qlikview.

Cuando la instalación de Qlikview server termine, el programa solicitará reinicio del servidor. Esto debe hacerse para que la funcionalidad de Qlikview quede completa. Ver figura 4.14.



Figura 4.14. Pantalla de finalización de instalación de Qlikview.

Una vez reseteado el servidor, en los servicios locales se deben iniciar los 5 servicios de QlikView server:

1. Qlikview directory service connector.
2. Qlikview distribution service.
3. Qlikview management service.
4. Qlikview server.
5. Qlikview webserver.

Estos servicios serán iniciados por el usuario de dominio especificado en la instalación. Ver figura 4.15.

Nombre	Descripción	Estado	Tipo de inicio
Qlikview Directory Service Connector	Keeps track ...	Iniciado	Automático
QlikView Distribution Service	The service t...	Iniciado	Automático
Qlikview Management Service	Hosts the M...	Iniciado	Automático
Qlikview Server	Hosts the Ql...	Iniciado	Automático
Qlikview Webserver	The web ser...	Iniciado	Automático
Recopilador de eventos de Windows	Este servicio...		Manual
Redirector de puerto en modo usuario de S...	Permite la re...		Manual
Registrador de configuración de Windows ...	WCNCSVC ...	Iniciado	Manual
Registro de eventos de Windows	Este servicio...	Iniciado	Automático
Registro remoto	Habilita usu...		Manual
Registros y alertas de rendimiento	Registros y a...		Manual

Figura 4.15. Pantalla de servicios del dominio en Qlikview.

Una vez iniciados los servicios, se podrá ingresar al QEMC (Consola de administración incluida en el paquete de Qlikview) y activar la licencia de tipo servidor, para eso se debe seleccionar la pestaña "System" y después seleccionar la pestaña "License", enseguida elegir el servidor de Qlikview. En la pestaña "Qlikview Server License" es necesario ingresar los datos solicitados y al finalizar hacer click en "Apply License". Ver figura 4.16.

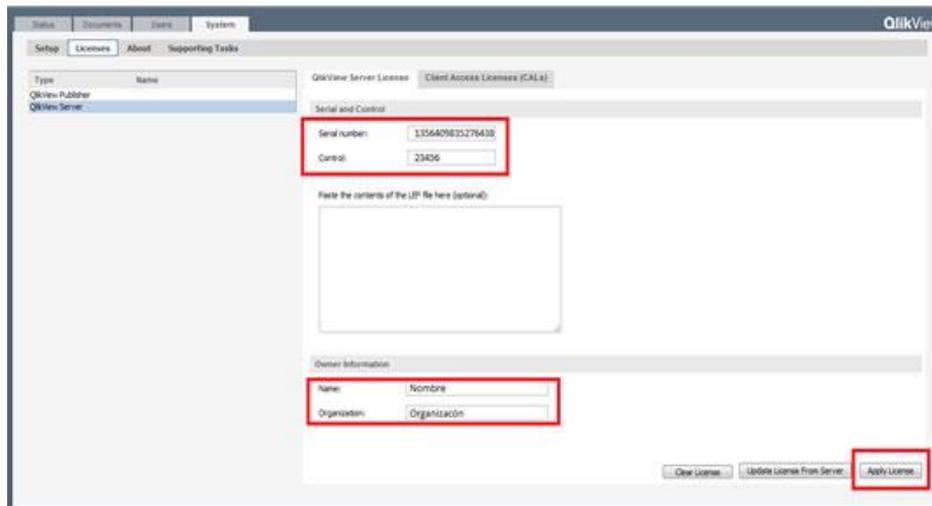


Figura 4.16. Pantalla de activación de licencia en Qlikview.

Como la aplicación se encuentra montada en un servidor de red se puede ejecutar desde cualquier dispositivo móvil que el administrador autorice de acuerdo a los perfiles que se manejen en la organización.

4.2. Documentación técnica

Antes de empezar a utilizar el sistema, es importante entender los términos y las convenciones tipográficas usadas en esta documentación. En esta sección explicamos algunos de los términos.

Convenciones generales

- La expresión "elegir" se utiliza para ejecutar un comando del menú en la barra de herramientas o en un diálogo.

- La expresión "seleccionar" se utiliza para marcar en una lista o en una hoja un objeto con el que quiere realizar su siguiente acción. También se utiliza para marcar valores en campos, haciendo selecciones lógicas entre los datos.
- Las listas enumeradas (p.ej. 1, 2, 3,...) indican procesos con uno o más pasos secuenciales.
- Las listas de referencia, como ésta, proporcionan información y no indican pasos secuenciales.

Convenciones del ratón

- También se puede usar un ratón de un solo botón o de varios botones en el sistema. Recomendamos el uso de un ratón de múltiples botones.
- Si su ratón es de múltiples botones, las descripciones de este documento asumen que el botón izquierdo de su ratón está configurado como botón principal y el derecho como botón secundario del ratón.
- "Apuntar a..." significa mover el ratón hasta colocar la punta del cursor encima del objeto en cuestión.
- "Click" significa apuntar al objeto en cuestión y luego pulsar e inmediatamente soltar el botón del ratón sin mover el ratón.
- "Doble-click" significa hacer rápida y sucesivamente click con el botón del ratón.
- "Click derecho" significa hacer click con el botón derecho del ratón.

Convenciones de teclado

- Los nombres de las teclas parecen en letra mayúscula pequeña, por ejemplo "Pulse INTRO".
- La tecla "RETURN" y la tecla "INTRO" ejecutan lo mismo en QlikView.
- El símbolo más "+" entre dos nombres de teclas indica que hay que pulsar ambas teclas a la vez por ejemplo "CTRL"+"S" significa que debe pulsar y mantener presionada la tecla "CTRL" al mismo tiempo que se pulsa la tecla "S".
- Un símbolo coma "," entre dos nombres de teclas indica que hay que pulsar las teclas sucesivamente.

La instalación del software utilizado en el sistema.

El paquete de instalación incluye un manual, una nota con un número de serie, un número de control y un acuerdo de licencia.

El software está diseñado para ejecutarse en un equipo compatible sobre Windows 98, Windows ME, Windows NT 4, Windows 2000, Windows XP, Windows Vista, Windows 7,

Windows 8 ó posterior. Para información más detallada acerca de los requerimientos del sistema se incluyen las siguientes tablas 4.2 y 4.3.

Sistema	32- bits (x86)	64 – bits (x64)
Sistema operativo	Windows XP professional Windows 7 u 8 Windows server 2003 Windows server 2012	Windows XP professional Edition Windows vista x64 Windows server x64 2003 Windows server x64 2012
Procesador	Se recomienda Intel Core I3 ó posterior	Se recomienda Intel Core 2 Duo o superior
Memoria	Mínimo 1 GB de acuerdo a los volúmenes de datos ¹	Mínimo 2 GB de acuerdo al volumen de datos se puede requerir más memoria. ¹
Capacidad del disco	Capacidad total necesaria para la instalación: 200 MB	Capacidad total necesaria para la instalación: 250 MB
Seguridad	Windows integrated Security o autenticación a medida ²	Windows integrated Security o autenticación a medida ²
Servidor WEB	Qlikview http server o Microsoft IIS 6.0 ó 7.0	Qlikview http server o Microsoft IIS 6.0 ó 7.0
Consola de gestión	Microsoft Internet Explorer Firefox ³	Microsoft Internet Explorer Firefox ³
.NET	3.5	3.5

¹ QlikView es una tecnología de análisis en memoria; los requisitos de memoria para los productos QlikView se relacionan directamente con la cantidad de datos que se analizan

² La autenticación a medida requiere QlikView Server Enterprise Edition

³ Otros navegadores web pueden funcionar con el cliente QlikView Analyzer AJAX, pero no han sido probados por QlikTech

Tabla.4.2. Requerimientos del sistema.

Soprote navegador/móvil	Portal Access Point QlikView	Qlikview para internet Explorer	Cliente Qlikview Java	Cliente QlikView AJAX	Qlikview para Iphone	Qlikview Java Mobile Edition
Prerrequisitos						
Qlikview server Small Business Edition	✓	✓	✓			
Qlikview server Enterprise Edition	✓	✓	✓	✓	✓	✓
Navegador						
Internet Explorer 7/8	✓	✓	✓	✓		
Firefox (windows, MAC)			✓	✓		
Safari (windows, MAC)			✓	✓		
Google Chrome (Windows)			✓	✓		
Apple iphone 2G y 3G ipod touch					Firmware 2.0 ó superior	

Tabla.4.3. Tabla de soportes de navegadores y móviles.

Al Iniciar el programa de instalación, realice lo siguiente:

1. Inserte el CD en la unidad CD-ROM, o memoria USB o dispositivo de almacenamiento en donde se encuentra el programa de instalación.
2. Si utiliza el CD, normalmente el programa de instalación (setup) se inicia sólo. Si no lo hiciese sólo, abra el Explorador de Windows, busque la unidad de CD (o memoria USB) y haga doble click en "Installer.exe".
Ahora se debería haber iniciado el programa de instalación.

Cuando se está ejecutando el programa de instalación

1. Elija su idioma en el primer diálogo que se abre y haga click en "Siguiente".

2. En el segundo diálogo, elija los elementos que desea instalar. Si desea leer el manual en formato pdf necesitará el programa Acrobat Reader. El Data Access Package contiene OLEDB, una interfaz necesaria para acceder a una base de datos general. Para poder acceder a los archivos de ayuda de Qlikview, se necesita un explorador de internet (versión 4.0 ó posterior). Sin embargo, a no ser que tenga una versión muy antigua de Windows, probablemente tenga ya un explorador de Internet instalado en su computadora. Cuando haya elegido las opciones deseadas, haga click en “Siguiente”.

3. Se lanzará el programa de instalación de Qlikview 11. En general, la opción que va a elegir es instalación en un Puesto (Usuario). Haga click en “Siguiente”.

4. Este paso le llevará al acuerdo de licencia. Después de leerlo, haga click en “Estoy de acuerdo” (si es este el caso).

5. El programa de instalación sugiere un directorio de destino para los archivos QlikView. Si en su computadora existe una versión anterior de Qlikview, el directorio de éste será el directorio por defecto. Si desea instalar Qlikview en otro directorio, haga click en “Examinar”, si no, haga click en “Siguiente”.

6. Los dos diálogos siguientes sugieren la instalación de unos archivos de backup y un directorio por defecto para éstos. Haga su selección, a continuación haga click en “Siguiente”.

7. Ahora le pide que seleccione los componentes que desea instalar. Haga su selección, luego haga click en “Siguiente”.

8. Finalmente, seleccione una carpeta de inicio apropiada para Qlikview. Haga click en “Siguiente”.

9. El programa instalará ahora los componentes Qlikview seleccionados en el directorio designado. Si ha elegido más temas en el 2º paso, éstos se instalarán una vez finalizada la instalación de Qlikview.

Serialización

Si está usando Qlikview Enterprise o Qlikview Professional el paso siguiente es realizar su copia de Qlikview.

1. Inicie Qlikview desde el menú Inicio. El cuadro de diálogo “Serialización” debe aparecer ahora. Es aquí donde debe introducir su número de serie, su número de control, su organización y su nombre de usuario.
2. Si no se ingresa la información de forma correcta, fallará la inicialización de Qlikview.
3. Si está usando Qlikview Analyzer no se necesita serialización. Sin embargo es posible adquirir un número de serie para este producto e introducirlo en la pestaña registro de las preferencias de usuario.
4. El Qlikview Analyzer registrado le dará acceso a archivos locales. Para más detalles se debe contactar al distribuidor del software.

Una vez instalada y serializada su copia de Qlikview, puede comenzar a utilizar el programa.

Instalar Qlikview en un servidor

La instalación en un servidor es útil si tiene muchas licencias y desea asegurar que todos los usuarios utilicen la misma versión de Qlikview. Cada usuario necesita un número de licencia distinto.

Para instalar Qlikview en un servidor, necesita el CD (o memoria USB o dispositivo de almacenamiento en dónde se encuentre el programa de instalación). No hacen falta los números de serie y números de control hasta iniciar la primera sesión Qlikview en un equipo cliente.

Haga lo siguiente:

1. Inicie el programa de instalación de Qlikview. (Vea más arriba en “Iniciar el programa de instalación”).
2. Elija “Instalación en Servidor” y siga las instrucciones. Ahora se instalarán Qlikview y un programa de instalación cliente en el servidor.

Instalar Qlikview en un equipo cliente

Para una instalación cliente de Qlikview, necesitará un número de serie con el correspondiente número de control para cada cliente que va a utilizar Qlikview.

Haga lo siguiente:

1. Inicie el programa de instalación Qlikview desde el servidor. Se hace igual que en la descripción anterior, con la diferencia que la ruta de acceso hacia el programa de instalación es x:\Qlikview\11\setup.exe o algo similar.
2. Siga las instrucciones. Qlikview se instalará en el equipo cliente.

Nota: La opción de instalar en un servidor sólo sirve para simplificar instalaciones en un área de red local con muchos usuarios Qlikview. No es una verdadera instalación cliente/servidor y no debe ser confundida con otras tecnologías de Qlikview.

OLE DB

Qlikview soporta la interfaz OLE DB para conexiones a fuentes de datos externas. Le permite acceder a muchas bases de datos mediante el OLE DB Provider de Microsoft para ODBC Drivers.

¿Cuándo es necesario la interfaz ODBC?

Es necesario tener la interfaz ODBC instalada si desea acceder a una base de datos general. La alternativa consiste en exportar datos desde la base de datos a un archivo en un formato legible para Qlikview.

Al ejecutarse el programa de instalación de Qlikview, puede elegir la instalación del paquete de acceso a datos (Data Access Package), el cual contiene ODBC.

Se pueden comprar drivers adicionales de proveedores de software. Algunos drivers se distribuyen gratuitamente.

Si lo que se pretende es usar ODBC para acceder a una base de datos relacional de muchos usuarios en un servidor de red, posiblemente necesitará software DBMS adicional (por ejemplo el SQL Net para ORACLE 7) que permite al cliente acceder a una base de datos en

el servidor. Contacte con su proveedor de DBMS para más información acerca del software requerido.

¿Es correcto el driver ODBC instalado?

Es posible que en su sistema ya tenga instalado un administrador ODBC. Para saber si está instalado o no, realice lo siguiente:

1. Haga click en el botón “Inicio” en la barra de tareas, a continuación elija “Configuración” y después “Panel de control”.
2. Si tiene un icono con el título “ODBC”, el administrador ODBC está instalado. Existen dos tipos, el ODBC y 32-bit ODBC.
3. Haga doble click en el icono “ODBC” correcto para averiguar que bases de datos están definidas.
4. Verifique en la pestaña “ODBC Drivers” los drivers instalados.

Si el administrador ODBC no está instalado, o si no encuentra el driver que necesita en la página de diálogo ODBC Drivers, instale el paquete de acceso a datos (Data Access Package) (que incluye ODBC) desde el programa de instalación QlikView o consiga el driver de un proveedor de software.

Crear fuentes de datos ODBC

Es necesario crear una fuente de datos ODBC para la base de datos a la que desea acceder. Esto se puede hacer durante la instalación de ODBC o en un nivel posterior.

Antes de comenzar con la creación de fuentes de datos, debe decidir si su fuente de datos va a ser de usuario o de sistema. A las fuentes de datos de usuario no se puede llegar hasta introducirse con el ID de usuario correcto. Si desea compartir sus fuentes de datos con otros usuarios, debe crear fuentes de datos de sistema. Al ejecutar Qlikview es imprescindible usar fuentes de datos de sistema.

1. Vuelva a abrir el cuadro de diálogo “ODBC Administrador de fuentes de datos”.
2. Abra la pestaña “User DSN” para crear fuentes de datos de usuario, o la pestaña “System DSN” si desea crear fuentes de datos de sistema.
3. Haga click en el botón “Añadir”, se abrirá el cuadro de diálogo “Añadir fuente de datos”, que le muestra una lista de los drivers ODBC instalados.

4. Si el driver ODBC correcto está en la lista, selecciónelo y haga click en el botón “Aceptar”. Se abrirá una caja de diálogo que corresponde al driver de base de datos seleccionado. Inserte el nombre de su fuente de datos y configure los parámetros necesarios. Cuando haya terminado haga click en “Aceptar”.

Hasta el momento ha aprendido a instalar de manera correcta Qlikview de acuerdo a sus necesidades.

Ahora pasemos a la ejecución del sistema que compete a esta documentación “Sistema de apoyo en la planeación de la programación de televisoras mexicanas”.

Si todo está bien el sistema debe ejecutarse sin ningún problema, pero ¿Qué pasaría si al consultar la información que genera en cada una de las pestañas no muestra los resultados esperados? ¿Qué hacer si no se muestra algún canal de televisión cargado en el sistema? ¿Cómo puedo saber que lo mostrado es lo correcto?

Para contestar todas estas preguntas a continuación se detallan soluciones posibles a los inconvenientes dando el panorama en donde buscar la solución.

Ahora bien ¿Qué hacer si no muestra algún canal de televisión cargado en el sistema?

Para resolver esto es muy sencillo, primero debe de asegurarse que en el archivo llamado canales.xls los nombres estén escritos de la manera correcta respetando la nomenclatura establecida. Es decir, seguimos la ruta donde se encuentran guardados los archivos .XLS que para este caso es: 01_DATA/XLS/CFG/canales.xls. Abrimos el archivo y procedemos a verificar los nombres uno por uno.

Éste es un error muy común que puede presentarse.

Recuerde que cada cambio que se realice en los archivos implica que hay que ejecutar y realizar la transformación nuevamente al sistema.

La respuesta a las otras dos preguntas tiene que mucho que ver con el contenido de la información de los archivos hora a hora y programación ya que estos archivos alimentan el sistema.

Para ello se recomienda revisar el siguiente apartado llamado pruebas de usuario final y revisar el apartado validación de datos y funcionalidad. En las cuales se presentan tablas que describen y demuestran la validación de la información.

4.3 Pruebas de usuario final

El objetivo de estas pruebas es verificar el correcto funcionamiento de los objetos entre las distintas pestañas que componen el sistema. Con la realización de éstas, es importante confirmar los requerimientos del usuario final para obtener la aceptación del mismo. A continuación se muestra un tabla con las pruebas realizadas al sistema.

Objetivo de las pruebas	Comprobar el funcionamiento del sistema.
-------------------------	--

Ítem	Descripción	Observaciones	Resultados Obtenidos	Comentarios
1	Revisar la fecha de actualización de la información contenida en el sistema.	Verificar el dato en la pantalla de inicio.	Fecha correcta.	Ninguno.
2	Verificar que la información en el sistema corresponda a los meses de enero, abril y julio del 2015.	Verificarlo en el objeto fecha de la pestaña de canales.	La información corresponde a los meses de enero, abril y julio de 2015.	Ninguno.
3	Verificar que existan los seis canales con su programación para un día seleccionado.	Utilizar como filtro la fecha del 3 enero del 2015 para validar la información.	Revisar el archivo 3 enero 2015.	Ninguno.
4	Validar que funcionan los mensajes de aviso de selección de parámetros para mostrar la información.	Selecciona dos de tres de los parámetros solicitados para ver que ocurre.	El mensaje de aviso aparece cuando cuando no se cumple la condición.	Ninguno.
5	Confirmar en la pestaña análisis de audiencia que las gráficas generadas sean iguales cuando se selecciona la misma variable para ambos filtros.	Seleccionar el share (Shr%) en ambas variables.	Programa las noticias de las 22 en DF.	Ninguno.

Tabla.4.4. Pruebas de usuario final.

6	Verificar que las fechas en la pestaña análisis de programas se muestren en forma ascendente.	Verificarlo en el objeto fecha de la pestaña análisis de programa.	Las fechas se muestra en forma ascendente.	Ninguno.
7	Verificar que todos los botones direccionen al hipervínculo correcto.	Verificarlo en todas las pestañas.	Botones en correcto funcionamiento.	Ninguno.
8	Verificar que en la pestaña canales al posicionar el cursor en un punto sobre el gráfico aparecen los valores de los parámetros: inicio, canal, programa y duración.	Verificarlo al posicionar el cursos en diferentes puntos el gráfico.	Información mostrada correctamente.	Ninguno.
9	Verificar que los colores correspondan con los canales cargados en el sistema.	Verificarlo con la lista de cotejo de colores.	Los colores corresponden con los canales.	Ninguno.
10	Verificar que los objetos permitan realizar búsquedas.	Verificarlo en cada uno de los objetos creados.	Se pueden realizar búsquedas en los objetos.	Ninguno.

Tabla.4.4. Continuación de tabla de pruebas de usuario final.

4.4 Capacitación

Objetivo

Capacitar a los usuarios finales en el uso del “Sistema de apoyo en la planeación de la programación en las televisoras mexicanas”. Esta capacitación permitirá que los usuarios amplíen su productividad y enriquezcan el potencial del sistema y para ello es necesario que éstos cuenten con los conocimientos básicos de la herramienta Qlikview.

Puntos a revisar en la capacitación:

- Introducción
 - Carga de archivos nuevos y/o actualización de los existentes.
 - Paso a paso del proceso de recarga de datos ejecutando el script de la extracción, la transformación y la ejecución del sistema.
 - Conceptos básicos de creación de objetos en Qlikview.
- Análisis de canales.
- Análisis de la audiencia.
- Análisis de programa.

➤ **Introducción**

- Carga de archivos nuevos y/o actualización de los existentes.

La información que se maneja (y se genera) en el sistema se almacena en carpetas con la estructura siguiente:

01_DATA contiene los archivos con la información extraída de la fuente origen y archivos QVD generados.

02_TRANS contiene los datos leídos de la carpeta 01_DATA para ser transformados (manipulados y adaptados) para generar el modelo asociativo.

03_APP contiene los archivos ejecutables de nuestro sistema.

Para poder incluir más archivos o incluso alguna actualización generada en último momento a los archivos ya cargados en el sistema, para esto es necesario dirigirse a la carpeta “01_DATA” dentro del directorio “XLS” abrir la carpeta “CFG” que contiene los archivos fuente. Dentro de esta última carpeta se procede a realizar los movimientos que se requieran por el usuario.

- Paso a paso del proceso de recarga de datos ejecutando el script de extracción, la transformación y la ejecución del sistema.

En el paso anterior nos encontrábamos dentro de la carpeta “01_DATA” dentro de “XLS”. Siguiendo dentro de esta carpeta abra la carpeta “QVW” y dé click en el archivo “Extracción IBOPE” como se muestra en la figura 4.17.

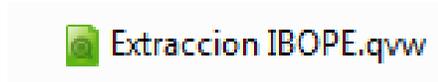


Figura 4.17. Icono de extracción en Qlikview.

Cuando termine de realizar el proceso anterior abra la carpeta “02_TRANS” para ejecutar ahora el archivo “.qvw” que realizará la transformación de los datos. Ver figura 4.18.

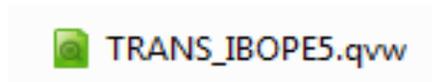


Figura 4.18. Icono de transformación en Qlikview.

Por último diríjase a la carpeta “03_APP” que contiene el archivo “.qvw” y ejecútelo dando click. Ver figura 4.19.



Figura 4.19. Icono de ejecución del sistema en Qlikview.

- Conceptos básicos de creación de objetos.

Dentro de la plataforma Qlikview se cuenta con una gran variedad de objetos que se pueden utilizar para poder organizar la información del sistema.

Para crear un objeto se debe dar click derecho sobre el área de trabajo, entonces aparecerá una lista de opciones en donde se debe seleccionar “**Nuevo Objeto de Hoja**” el cual mostrará un listado de los tipos de objetos existentes, como el generado en la figura 4.20.

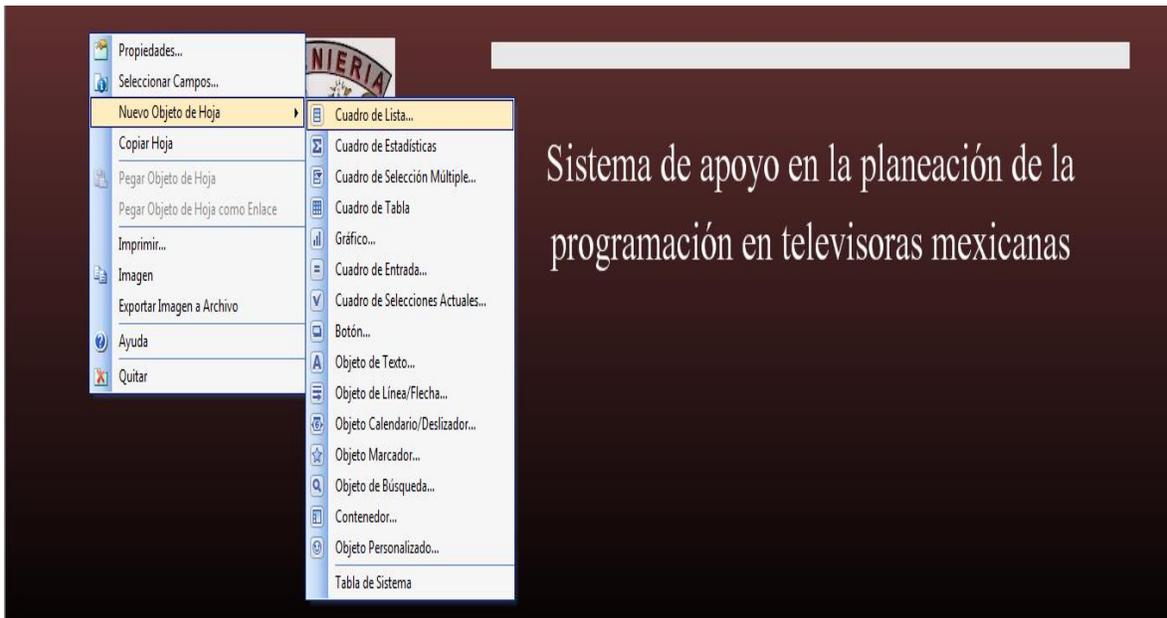


Figura 4.20. Imagen de creación de nuevos objetos.

Para la creación del sistema se utilizaron los siguientes tipos de objetos:

- ❖ Objetos de texto.
- ❖ Objetos de cuadro de selección múltiple.
- ❖ Objetos de gráfico.
- ❖ Objetos de cuadro de listas.

Ejercicio:

Abrir el sistema y generar una nueva pestaña:

- a) Dar click en el botón “Añadir hoja”.
- b) En la nueva hoja generada insertar un objeto de texto e introducir lo siguiente:
“Bienvenidos al curso de capacitación para la planeación de la programación de las televisoras mexicanas”. Ver figura 4.21.

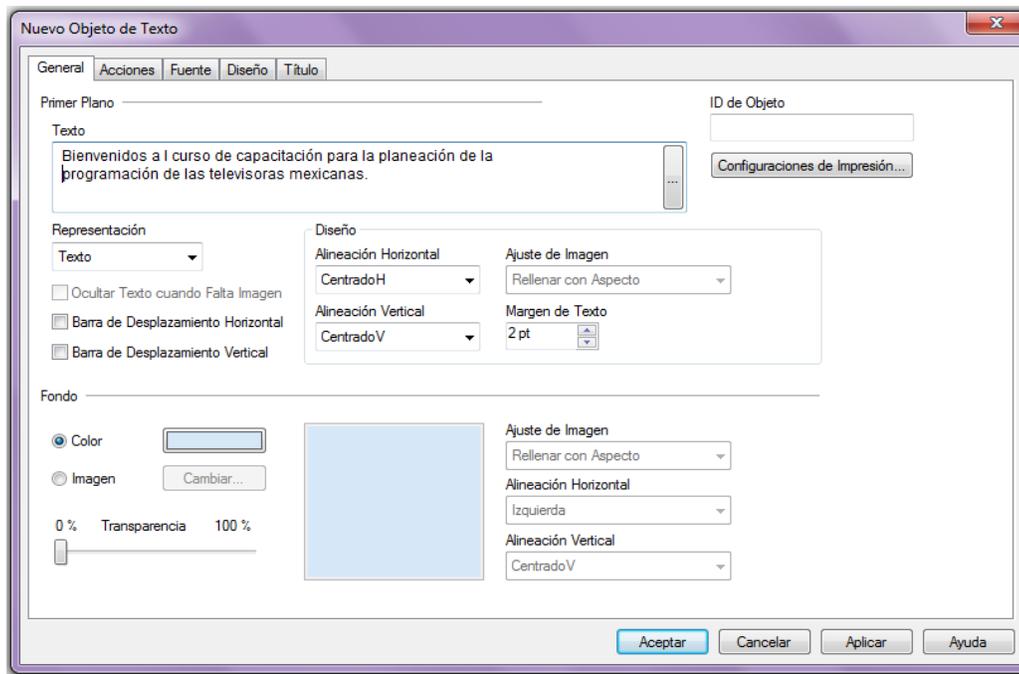


Figura 4.21. Ventana de propiedades del objeto texto.

En seguida realiza los cambios necesarios para que el texto se vea como la figura 4.22.

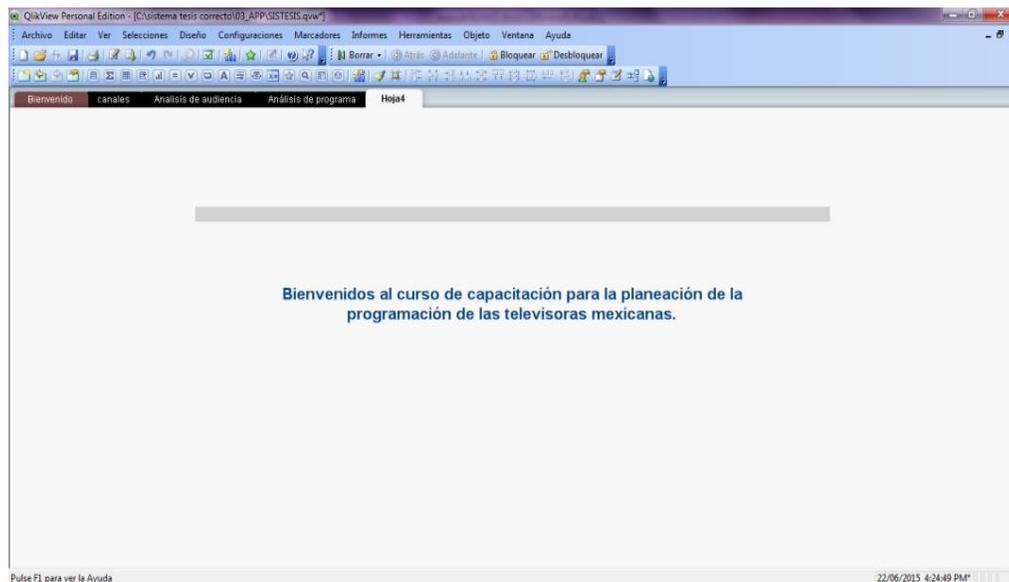


Figura 4.22. Imagen del ejercicio terminado.

Los pasos necesarios para cumplir con el objetivo del ejercicio son los siguientes:

- a) En la pestaña “General” modificar la transparencia al 100%.
- b) En la pestaña “Fuente” asignar el formato al texto: color, tamaño, tipo de letra, etc.
- c) En la pestaña “Título” activar la casilla “Mostrar título” para poder manipular el objeto de texto.
- d) Modificar los márgenes para que se aprecie todo el texto.

➤ **Análisis de canales**

NOTA: Inicialmente se abrirá la pestaña de “Bienvenido” lo que indica que es posible empezar a interactuar con el sistema, como la muestra la figura 4.23.

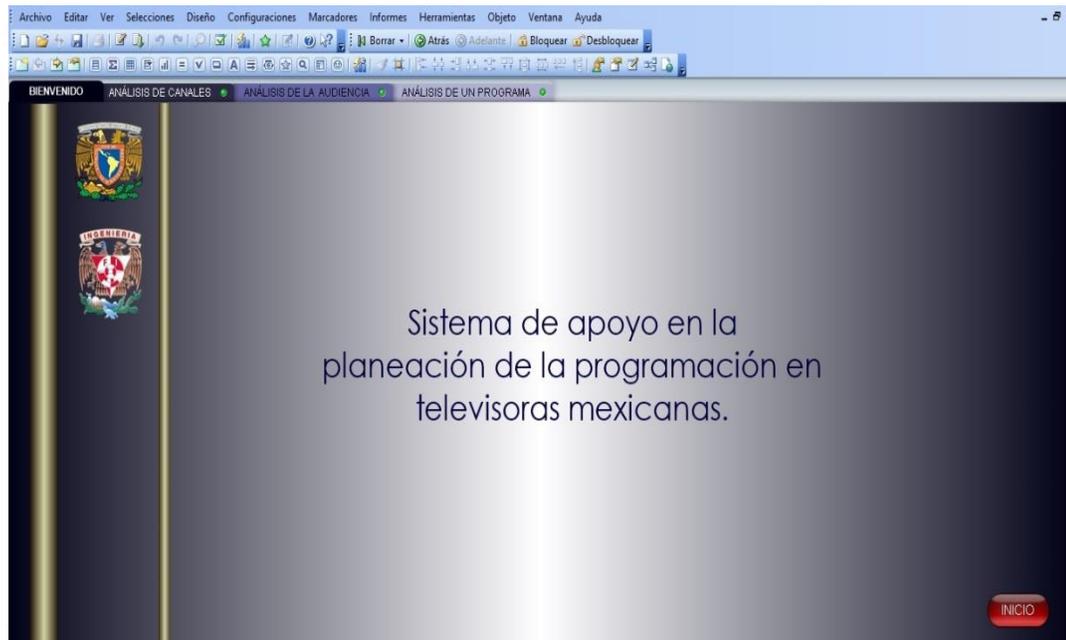


Figura 4.23. Pantalla de Bienvenida.

- a) La figura 4.23. muestra el logo que representa a la organización que creó el sistema.
- b) La fecha actual de la última carga.
- c) El nombre del sistema u organización.

A continuación se debe abrir la pestaña “Análisis de canales” que muestra una serie de objetos y textos que permiten seleccionar los parámetros para el análisis de la información que será mostrada en los gráficos en el sistema. Ver figura 4.24.

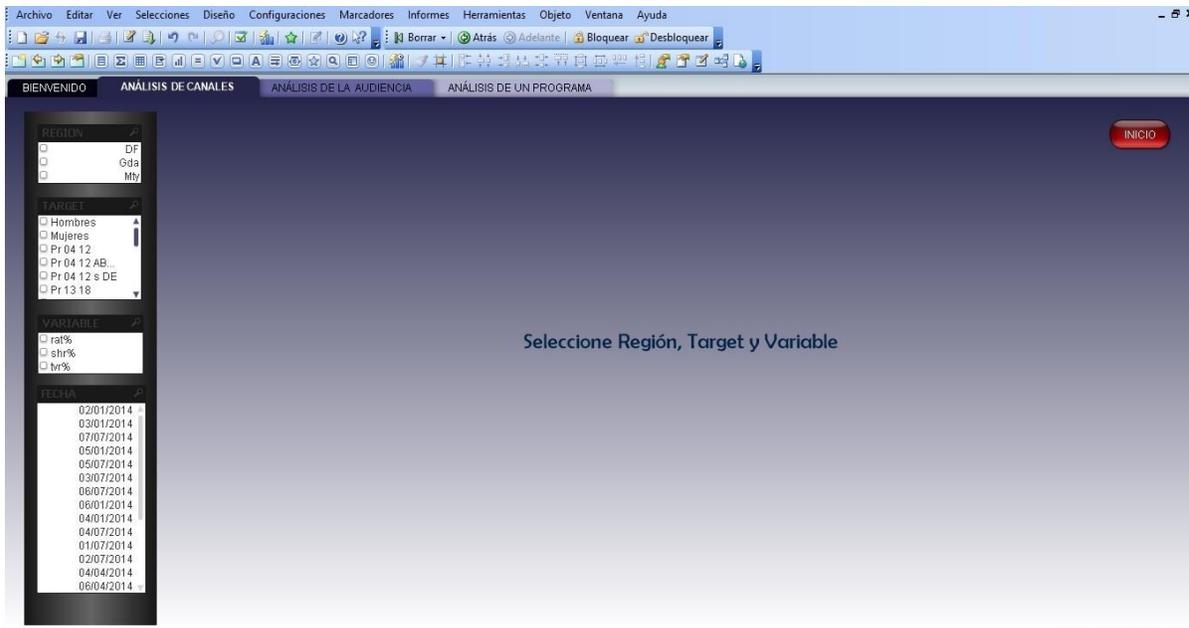


Figura 4.24. Pantalla de “Análisis de canales”.

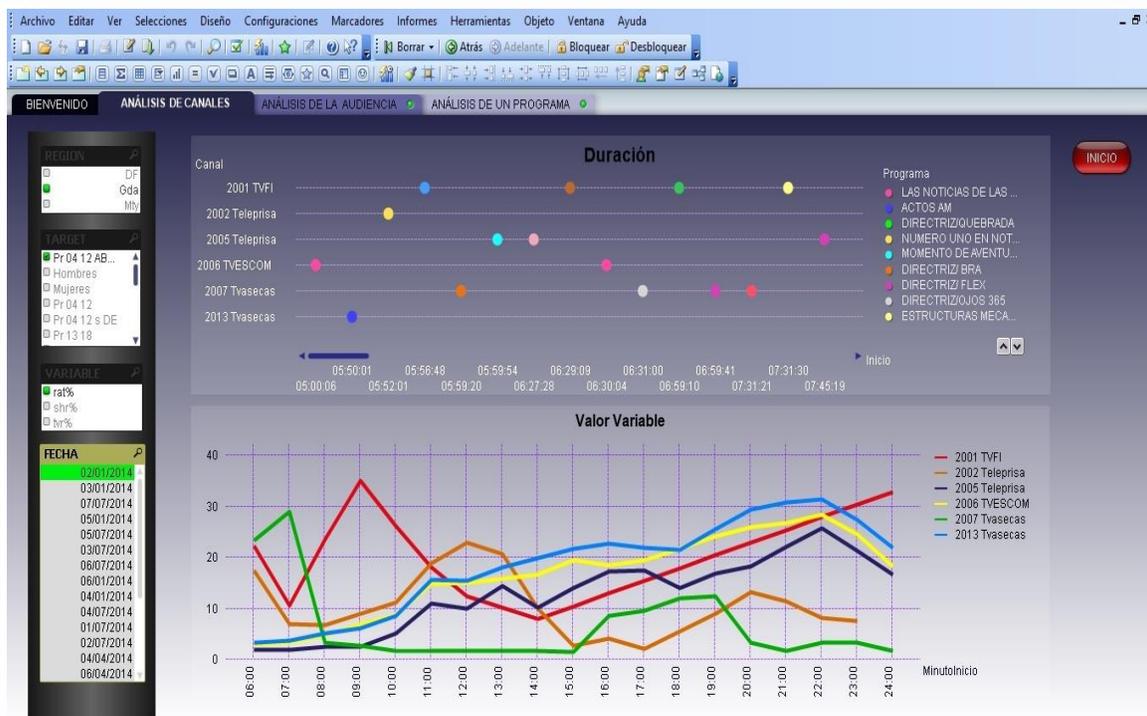


Figura 4.25. Pestaña de “Análisis de canales” terminada.

A la izquierda de la pantalla se encuentran los objetos en los cuales es posible seleccionar los 5 parámetros (región, target, variables, fecha y canal) requeridos para que las gráficas presenten los valores. Ver figura 4.26.

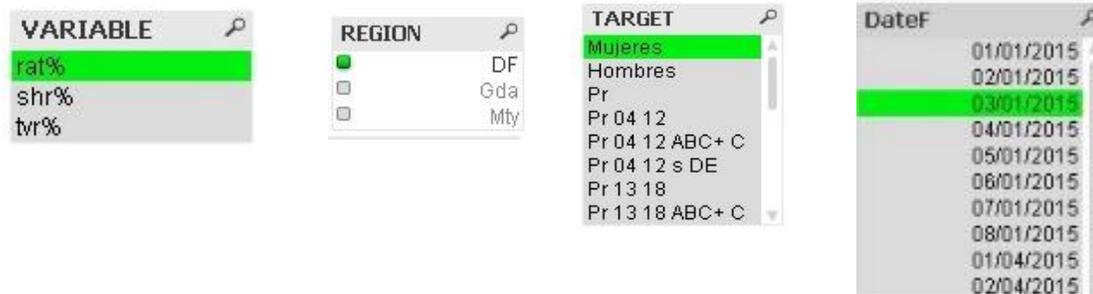


Figura 4.26. Objetos de selección de los parámetros.

La información seleccionada será mostrada en dos gráficos:

1. La primera muestra la programación transmitida a lo largo del día en los canales cargados permitiendo la comparación entre ellos y así deducir su comportamiento y tomar decisiones.
2. La segunda gráfica muestra el valor de la variable seleccionada en los canales de televisión hora por hora.

Ejercicio:

En la pestaña “ANÁLISIS DE CANALES” comparar los siguientes parámetros:

La programación de las televisoras 2001 TVFI contra 2006 TVESCOM del día 04/01/2015.

REGIÓN: DF.

TARGET: Hombres .

VARIABLE: rat%.

El sistema debe de mostrar los datos siguientes. Ver figura 4.27.

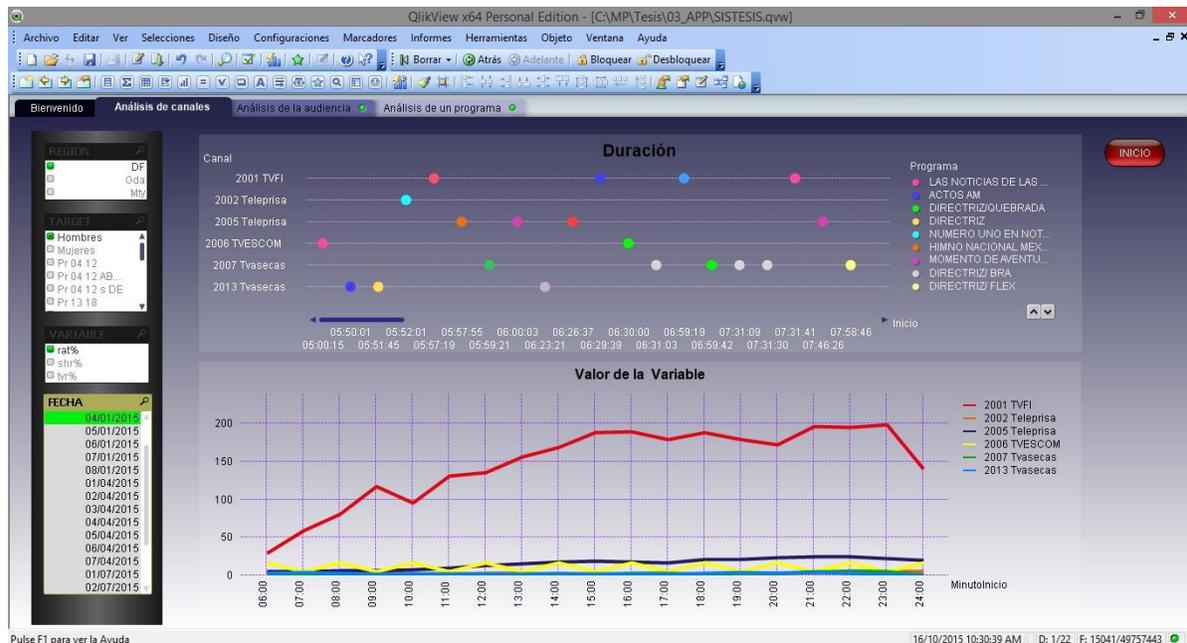


Figura.4.27. Imagen del ejercicio terminado.

➤ Análisis de la audiencia

Al abrir la pestaña “Análisis de la audiencia” se muestra una serie de objetos y textos que indican que parámetros se deben seleccionar para poder hacer un filtro de la información que mostrarán los gráficos en el sistema. Los parámetros a seleccionar son los siguientes:

REGIÓN: GDA.

PROGRAMA: ACTOS NOCHE .

V1: shr%.

V2: rat%.

Al seleccionar los parámetros indicados se mostrará lo siguiente en la figura 4.28.

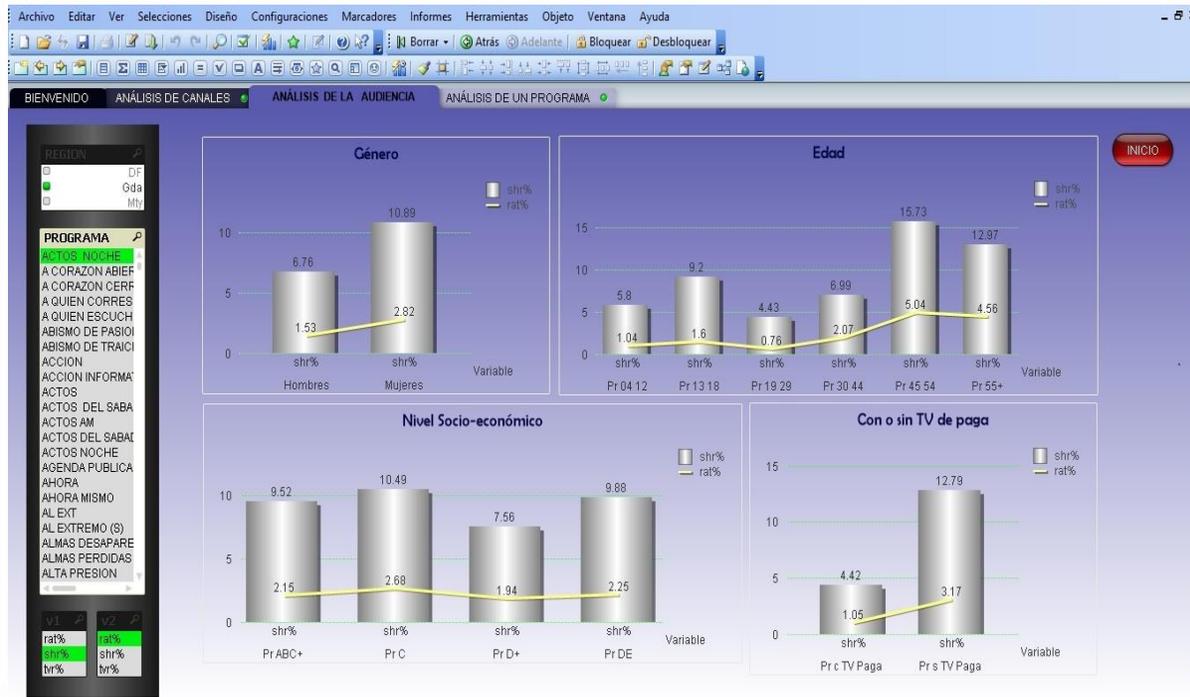


Figura 4.28. Pestaña “ANÁLISIS DE LA AUDIENCIA”.

La información seleccionada será mostrada en cuatro gráficos:

1. Cada uno de los gráficos representa a una muestra de la población en la cual podremos analizar el comportamiento por: “Género”, “Edad”, “Nivel Socio-económico” y personas “Con o sin TV de paga”.

Ejercicio:

En la pestaña “ANÁLISIS DE AUDIENCIA” seleccionar los siguientes parámetros y observar los resultados. Ver figura 4.29:

REGIÓN: GDA.

PROGRAMA: ACTOS DEL SÁBADO.

V1: rat%.

V2: shr%.

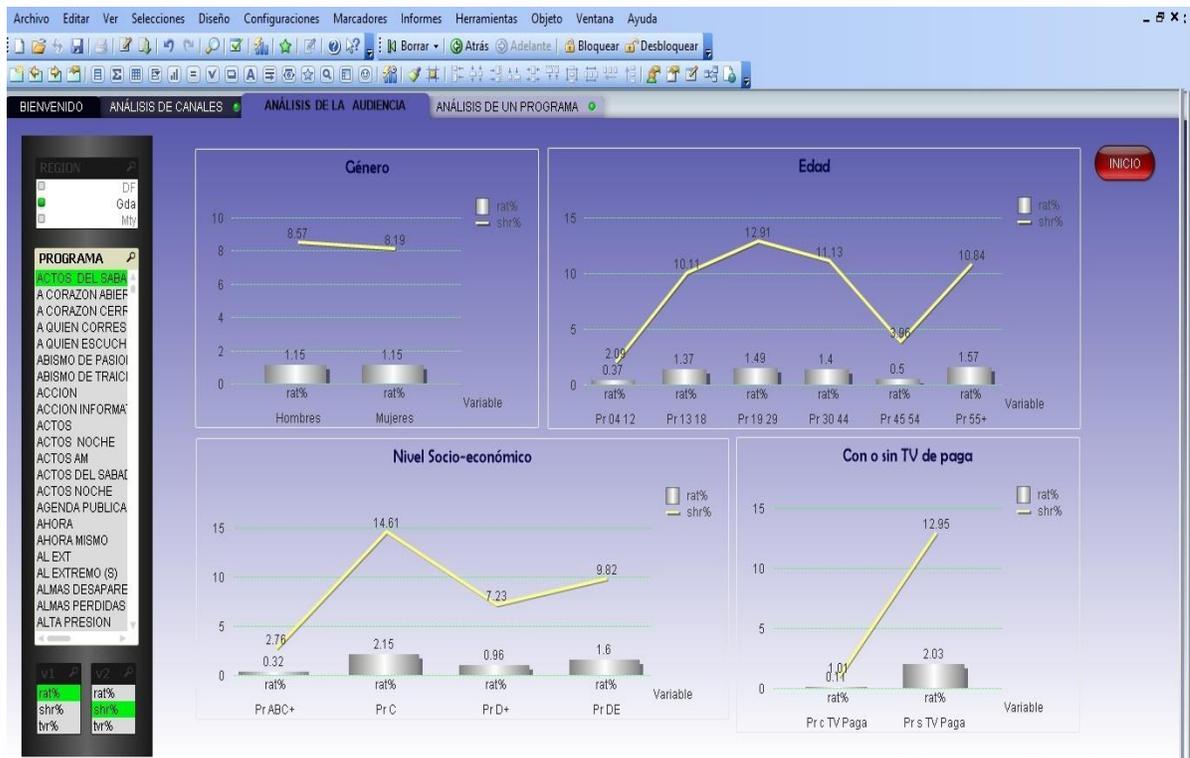


Figura. 4.29. Imagen del ejercicio terminado.

➤ Análisis de un programa

Al abrir la pestaña “Análisis de un programa” se muestra una serie de objetos y textos que indican que parámetros se deben seleccionar para poder hacer un filtro de la información que mostrarán las gráficas en el sistema. Los parámetros a seleccionar son:

REGIÓN: GDA.

PROGRAMA: ACTOS.

TARGET:Hombres.

VARIABLE: rat%.

Al seleccionar los parámetros indicados se tendrá una imagen como se observa en la figura 4.30.

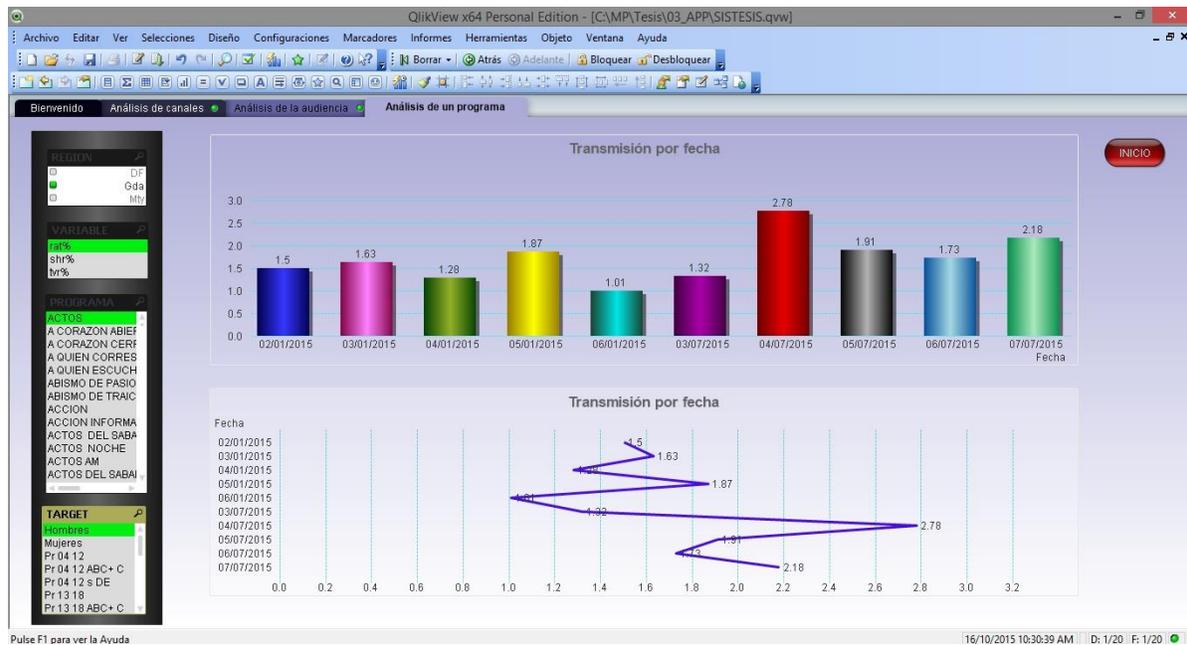


Figura 4.30. Pestaña “Análisis de un programa”.

Ambos gráficos muestran el comportamiento de la audiencia de un cierto programa a lo largo de varias transmisiones. Mostrando las fechas en orden ascendente y en varios colores.

Ejercicio:

En la pestaña “ANÁLISIS DE UN PROGRAMA” seleccionar los siguientes parámetros:

REGIÓN: GDA.

TARGET: Mujeres.

PROGRAMA: CUADRO AMARILLO.

VARIABLE: rat%.

Es necesario analizar la información mostrada por el sistema y comparar con el resultado de la figura 4.31.

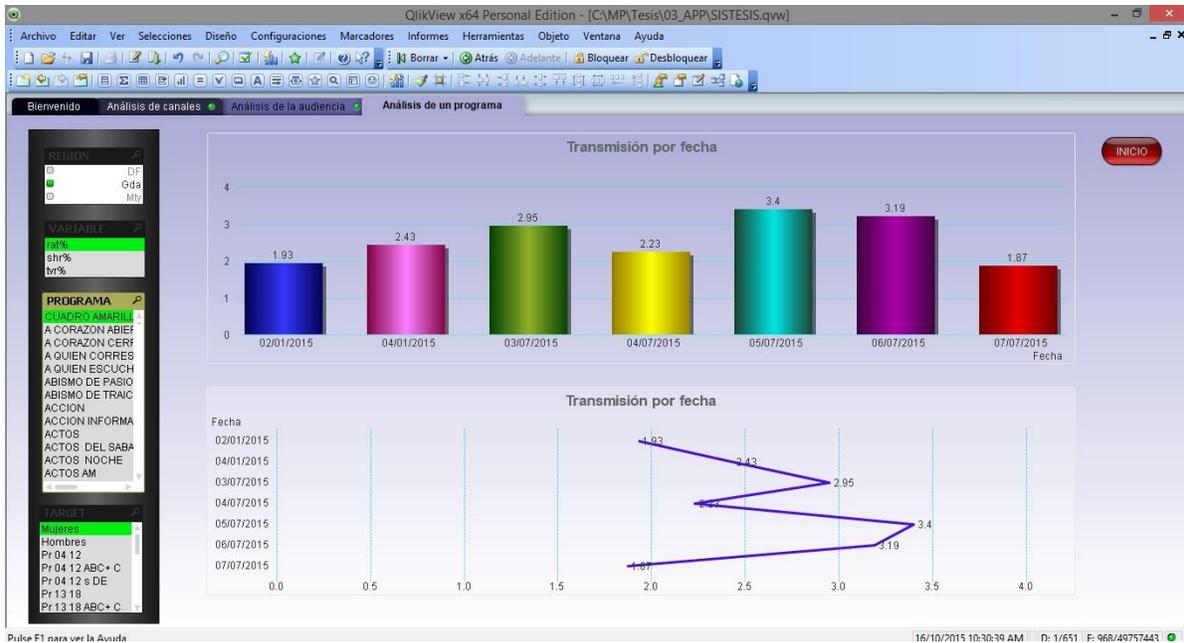


Figura. 4.31. Imagen del ejercicio terminado.

4.5 Documento de cierre

Al finalizar el desarrollo de un sistema se elabora un “Documento de cierre” como el propuesto en el **ANEXO C** de esta tesis. Este documento tiene como objetivo formalizar la conclusión de actividades y término de responsabilidades entre la parte usuaria y el área de desarrollo, confirmando mediante aprobación que se cumplieron los objetivos definidos.

CONCLUSIONES

El sistema desarrollado y propuesto puede ser implementado en cualquiera de las televisoras que operan en la república mexicana, siempre y cuando se cuente con los recursos de hardware y software necesarios para su operación según las especificaciones técnicas proporcionadas por Qliktech.

En la actualidad muchas de las empresas incluyendo las televisoras, disponen de una enorme cantidad de datos almacenados en grandes bases de datos que van creciendo fenomenalmente y su complejidad de procesamiento se incrementa dificultando el análisis y presentación.

Cuando se desarrolla un sistema de software es de suma importancia saber identificar las necesidades del usuario pero sobre todo contar con una comprensión clara de las mismas ya que de no ser así el sistema puede no ser útil y como consecuencia se generan gastos innecesarios tanto de tiempo como de dinero; por esta razón debemos dedicar tiempo suficiente para el análisis de los requerimientos de datos y los requerimientos de funcionalidad los cuales deben ser completos, realistas y verificables. Es necesario entender el problema de los usuarios en su cultura, con su lenguaje, identificar al usuario clave y construir un sistema de software que realmente resuelva sus necesidades.

El modelo asociativo es el corazón y parte fundamental del sistema de software propuesto ya que es la consolidación de las variadas fuentes de información a las que tenemos acceso, nos permitirá visualizar los datos desde muchas perspectivas debido a que todos los valores de campos de todas las tablas en el modelo están asociados entre sí de forma automática en base solamente al nombre de los campos.

En el momento en que se trabaja en la construcción del modelo asociativo, es preferible invertir en el script de carga y no tanto en la formulación de expresiones dentro de los objetos, con ello se gana velocidad frente al usuario teniendo el dato ya pre-calculado y se soportan los diferentes análisis requeridos por los usuarios.

Si contamos con nombres claros de los campos y con sólidos conocimientos de programación, podremos definir una sólida arquitectura asociativa que evitará que se sobrecargue el sistema y así se aprovecha cada fragmento de hardware y cada byte de la memoria facilitando el procesamiento de los datos y el mantenimiento del sistema.

Un factor muy importante es el buen dimensionamiento de las necesidades de infraestructura de hardware el cual se determina principalmente en función de la cantidad de registros que se van a procesar. En el sistema el tiempo de carga de los datos

dependerá de los recursos técnicos del equipo de cómputo en donde se ejecute y dicha carga será definida por cada usuario.

La versión gratuita del software utilizado ofrece todas las bondades para crear y diseñar sistemas pero solo de manera local, esto impide que se puedan visualizar las aplicaciones diseñadas con el mismo software en otros equipos de cómputo. Es necesario entonces adquirir una licencia cuyo tipo dependerá de los intereses de la organización que desee utilizar el sistema. Una vez que se cuente con la licencia se puede tener acceso vía web a los demos creados para diversos campos profesionales lo que permite ampliar las ideas de diseño y reutilizar el código en otras aplicaciones.

El sistema desarrollado y propuesto ofrece respuestas visuales mediante objetos gráficos que atraen la atención del usuario, es un sistema ampliable que permite anexar fuentes de información, anexar pestañas y objetos según los intereses de cada usuario con la posibilidad de mejorar su desempeño en el futuro y garantizar el éxito de la operación.

Como parte del aprendizaje adquirido en la realización del presente trabajo podemos comentar que confirmamos la importancia de contar con los conocimientos técnicos para ofrecer alternativas de mejora a las empresas quienes siempre buscan reducir los costos de operación, en este caso por ejemplo podemos considerar que el sistema reducirá el tiempo que las autoridades de la organización deben esperar para contar con la información actualizada sobre su empresa, este tiempo impactará en la toma de decisiones y por ende en los resultados, por otro lado, el personal que utilice el sistema será más productivo al dedicar un mayor tiempo al análisis de la información la cual estará siempre disponible y cuyo acceso no tendrá intermediarios.

Durante el desarrollo de la presente tesis reafirmamos algunos conocimientos adquiridos en la facultad en los años de formación, también tuvimos un acercamiento con nuevas áreas que no conocíamos hasta entonces, de las que obtuvimos mucho aprendizaje (como la medición de variables de audiencia de televisión) y que nos ampliaron el panorama sobre el campo profesional.

Confirmamos la importancia de trabajar en equipo sin importar el número de participantes; nos enfrentamos a dificultades principalmente de horario pero buscamos soluciones ya que perseguimos un mismo objetivo.

Finalmente confirmamos que el objetivo de esta tesis se cumplió de forma satisfactoria.

ANEXO A

Código para el proceso de extracción de datos

Obtenemos todos los archivos contenidos en la carpeta

Archivos Programación:

```
LOAD Distinct FileName() as FileName,
      FileTime() as filetime
```

```
FROM
```

```
[..\CFG\TablasProgramacion\*.xlsx]
(ooxml, embedded labels);
```

```
// Obtenemos el máximo número de fila
```

```
Let i=0;
```

```
let varMaxNumRowP = NoOfRows('ArchivosProgramacion');
```

```
set vprimeraCargaDatosP = '0';
```

```
do while i < $(varMaxNumRowP)
```

```
// Obtenemos el nombre de archivos....
```

```
let varNameArchivoP = Peek('FileName', i, 'ArchivosProgramacion');
```

```
TRACE 'varNameArchivoP=' & '$(varNameArchivoP)';
```

```
for each vSheet in 'DF','Gda','Mty'
```

```
ProgramaTEMP:
```

```
CrossTable(VariableTarget, ValorVariable, 10)
```

```
LOAD * FROM
```

```
[..\CFG\TablasProgramacion\$(varNameArchivoP)]
```

```
(ooxml, embedded labels, table is $(vSheet));
```

```
programa:
```

```
load *,
```

```
replace ( replace('$(vSheet)', '[,]', ',' ) as Region
```

```
resident ProgramaTEMP;
```

```
drop table ProgramaTEMP;
```

```
next;
```

```
i=i+1;
```

```
loop;
```

```
if ( $(vprimeraCargaDatosP) = '0') then
```

```
store programa into [..\QVD\programacion.qvd] (qvd);
```

```
vprimeraCargaDatosP = '1';
```

```
TRACE 'primera Carga Datos ';
```

```
drop table programa;
```

```
ELSE
```

```
CONCATENATE (programa) load * from [..\QVD\programacion.qvd] (qvd);
store programa into [..\QVD\programacion.qvd] (qvd);
drop table programa;
ENDIF
```

```
//Salida del Script;
```

El código para leer los archivos de MinutoXMinuto es el siguiente:

```
// Obtenemos todos los archivos contenidos en la carpeta
```

```
ArchivosMinutoAMinuto:
```

```
LOAD Distinct FileName() as FileNameMin,
      FileTime() as filetimeMin
FROM
[..\CFG\MinutoAMinuto\*.xlsx]
(ooxml, embedded labels);
```

```
// Obtención del máximo número de fila
```

```
Let m=0;
let varMaxNumRowM = noofrows('ArchivosMinutoAMinuto');
set vprimeraCargaDatosM = '0'
do while m < $(varMaxNumRowM)
```

```
// Obtiene el nombre de los archivos....
```

```
let varNameArchivoM = Peek('FileNameMin', m,'ArchivosMinutoAMinuto');
TRACE 'varNameArchivoM=' & '$(varNameArchivoM)';
```

```
for each vSheet in 'DF','Gda','Mty'
```

```
MinutoAMinutoTemp:
```

```
LOAD * FROM
[..\CFG\MinutoAMinuto\$(varNameArchivoM)]
(ooxml, embedded labels, table is $(vSheet));
```

```
let vFechaTransmision = FieldName ( 1 , 'MinutoAMinutoTemp');
```

```
MinutoAMinutoTempRow:
```

```
LOAD
RecNo() as numRow,
replace ( replace('$(vSheet)', '[' , ']' ) as Dominio,
'$(vFechaTransmision)' as Fecha,
['$(vFechaTransmision)'] as Minuto,
*
resident MinutoAMinutoTemp;
```

```
drop table MinutoAMinutoTemp;
```

```

// Obtención de los canales.

CanalesTemp:
load Minuto as Canal,
numRow as RowCanal
Resident MinutoAMinutoTempRow
where not wildMatch( Minuto,'*:*')
      and not wildMatch(Minuto,'*/*')
and not wildMatch(Minuto,' ');

// Eliminar Columnas de la tabla MinutoAMinutoTempRow
DROP FIELD [$(vFechaTransmision)];

Canales:

NOCONCATENATE
LOAD *RESIDENT CanalesTemp ORDER BY RowCanal ASC;
DROP TABLE CanalesTemp;

// Obtención del número de canales
let numCanales = noofrows('Canales');

let z=0;
let vnumrow0 = "";
let vNombre0 = "";

If $(numCanales) > 0 then

// Generación de variables de intervalo.

do while z < $(numCanales)
if(z=0) then
vnumrow0= Peek('RowCanal',z,'Canales');
vNombre0= Peek('Canal',z,'Canales');
else
vnumrow0= '$(vnumrow0)' & ',' & Peek('RowCanal',z,'Canales');
vNombre0= '$(vNombre0)' & ',' & Peek('Canal',z,'Canales');
end if
TRACE ... TEST SUCCEEDED: Canales file records ... ;
z=z+1;
loop;

Else
TRACE ... TEST SUCCEEDED: Canales file has 0 records ... ;
End if
trace "vnumrow0=" $(vnumrow0);
trace "vNombre0=" $(vNombre0);

```

```

DROP table Canales;
// Agregar columna de canal

let numCanalesDat = $(numCanales) ;
let numNombre=1;
let minRow=1;
let maxRox=2;

if ($(numCanales) > 0) then

if( Len(vnumrow0) > 0 and Len(vNombre0) > 0 ) then

for a=1 to $(numCanalesDat)
if(a = $(numCanalesDat) ) then
MinutoAminuto2:
load
subfield('$(vNombre0)', ',' ,$(numNombre)) as Canal,
resident MinutoAMinutoTempRow
where numRows > subfield('$(vnumrow0)', ',' ,$(minRow));
else
MinutoAminuto2:
load
subfield('$(vNombre0)', ',' ,$(numNombre)) as Canal,
*
resident MinutoAMinutoTempRow
where numRows > subfield('$(vnumrow0)', ',' ,$(minRow)) and numRows < subfield('$(vnumrow0)', ',' ,$(maxRox)) ;
ENDIF

minRow = $(minRow)+ 1;
maxRox = $(maxRox) + 1;
numNombre = $(numNombre) +1;
x=a+1;
next
ENDIF
ENDIF
drop table MinutoAMinutoTempRow;

MinutoAminuto:
CrossTable(Target, Data, 5)
LOAD *
Resident MinutoAminuto2;
drop table MinutoAminuto2;
next; // cierra for de vSheet
m=m+1;
loop;

if $(vprimeraCargaDatosM) = '0') then
store MinutoAminuto into [..\QVD\MinutoAminuto.qvd] (qvd);
vprimeraCargaDatosM = '1';

```

```
TRACE 'primera Carga DATos MinutoAMinuto';  
drop table MinutoAminuto;  
ELSE
```

```
CONCATENATE (MinutoAminuto) load * from [..\QVD\MinutoAminuto.qvd] (qvd);  
store MinutoAminuto into [..\QVD\MinutoAminuto.qvd] (qvd);  
drop table MinutoAminuto;  
ENDIF
```

```
drop Fields FileName, filetime,FileNameMin,filetimeMin;
```

```
//Salida del Script;
```

Generamos un archivo en formato .qvw que nos permitirá validar los datos extraídos. El código que se utiliza para esta validación es el siguiente:

Programacion:

```
LOAD F1,  
  NIns,  
  Channel,  
  Description,  
  Date,  
  [Start Time {Av(Tm)}],  
  [End Time {Av(Tm)}],  
  [Length {Sum}],  
  TimeBand,  
  VariableTarget,  
  ValorVariable,  
  Region  
FROM  
[..\QVD\programacion.qvd]  
(qvd);  
exit Script;
```


ANEXO B

Código para el proceso de transformación de datos.

```
SET ThousandSep='';  
SET DecimalSep='.';  
SET MoneyThousandSep='';  
SET MoneyDecimalSep='.';  
SET MoneyFormat='$#,##0.00;-$#,##0.00';  
SET TimeFormat='hh:mm:ss TT';  
SET DateFormat='DD/MM/YYYY';  
SET TimestampFormat='DD/MM/YYYY hh:mm:ss[.fff] TT';  
SET MonthNames='ene.;feb.;mar.;abr.;may.;jun.;jul.;ago.;sep.;oct.;nov.;dic.';  
SET DayNames='lun.;mar.;mié.;jue.;vie.;sáb.;dom.';
```

```
CnalesTV:  
LOAD IdCanal,  
     IBOPE,  
     DivisonFilmica,  
     Ingresos as CanIngresos,  
     NombreCanal  
FROM  
[..\01_DATA\XLS\CFG\Canales.xls]  
(biff, embedded labels, table is Sheet1$);
```

```
// mapeo con Nombre de canal
```

```
CanalMinuto:  
Mapping  
load  
NombreCanal as Canal,  
IdCanal as IdCanal  
resident CnalesTV;
```

```
// Catálogo Canal.
```

```
Canal:  
load  
NombreCanal as NombreCanal,  
IdCanal as IdCanalTrans,  
CanIngresos  
Resident CnalesTV;
```

```
store Canal into [..\QVD\Canal.qvd](qvd);  
drop table CnalesTV;
```

ProgramacionTemp:

```

LOAD F1,
    NIns,
    Channel as Canal,
    Description,
    Date,
    [Start Time {Av(Tm)}],
    [End Time {Av(Tm)}],
    [Length {Sum}],
    TimeBand,
    VariableTarget,
    ValorVariable,
    Region
FROM
[..\..\01_DATA\XLS\QVD\programacion.qvd]
(qvd);

```

ProgramacionTemp1:

```

load
trim( if( (Len(Trim(Canal)) <> 0) and (index( Canal, '-' ) > 0) , left( Canal, num(len( subfield(Canal, '-',1) ) -
6 ) ) , Canal ) ) as Canal,
Description,
Date,
[Start Time {Av(Tm)}] as PrgTiempoInicio,
[End Time {Av(Tm)}] as PrgTiempoFin,
trim( if( (Len(Trim(VariableTarget)) > 0) , replace( VariableTarget,subfield(VariableTarget, '-',1)," ) , " ) ) as
PrgTarget,
trim(if( (Len(Trim(VariableTarget)) > 0) , subfield(VariableTarget, '-',1)," ) ) as PrgVariable,
ValorVariable as PrgValorVariable,
Region as PrgDominio
resident ProgramacionTemp;

```

drop table ProgramacionTemp;

ProgramacionTemp2:

```

load

PrgDominio as Dominio,
ApplyMap('CanalMinuto',Canal) AS IdCanal,
Date,
PrgVariable as Variable,
Description,
PrgTiempoInicio,
PrgTiempoFin,
PrgTiempoInicio as PrgTiempoInicioT,
PrgTiempoFin as PrgTiempoFinT,
PrgTiempoInicio as PrgTiempoInicioT1,
PrgTiempoFin as PrgTiempoFinT1,
PrgTarget as Target,

```

```

PrgValorVariable
resident ProgramacionTemp1;

drop table ProgramacionTemp1;

// Se crea la llave
Programacion:
load
( upper(Dominio) & IdCanal & Date & Target & Variable & PrgTiempoInicio & PrgTiempoFin) as
keyProgramacion,
( upper(Dominio) & IdCanal & Date & Target & Variable ) as keyPrograMin,
( upper(Dominio) & IdCanal & Date & Target & Variable ) as keyPrograMin2,
*
resident ProgramacionTemp2;

drop table ProgramacionTemp2;

//Salida del Script;

MinutoAMinutoTemp:
LOAD Canal,
      Dominio,
      Fecha,
      Minuto,
      Target,
      Data
FROM
[..\..\01_DATA\XLS\QVD\MinutoAminuto.qvd]
(qvd);

HoraAHoraTemp1:
load
      Dominio,
      trim( if( (Len(Trim(Canal)) <> 0) and (index( Canal, '-' ) > 0) , left( Canal, num(len( subfield(Canal, '-',1) ) -
6 ) ) , Canal ) ) as Canal, Fecha,
      trim(if( (Len(Trim(Minuto)) > 0) , subfield(Minuto, '-',1)," )) as MinutoInicio,
      trim(if( (Len(Trim(Target)) > 0) , subfield(Target, '-',1)," )) as MinVariable,
      trim( if( (Len(Trim(Target)) > 0) , replace( Target,subfield(Target, '-',1)," )," )) as MinTarget,
      Data as MinValorVariable
resident MinutoAMinutoTemp;

drop table MinutoAMinutoTemp;

HoraAHoraTemp2:
load

      Dominio,

```

```

ApplyMap('CanalMinuto',Canal) AS IdCanal,
Fecha as Date,
MinTarget as Target,
MinutoInicio,
MinVariable as Variable ,
MinValorVariable
resident MinutoAMinutoTemp1;

drop table MinutoAMinutoTemp1;

// Creación de llave

HoraAHoraTemp3:
load
( upper(Dominio) & IdCanal & Date & Target & Variable ) as keyPrograMin,
( upper(Dominio) & IdCanal & Date & Target & Variable ) as keyPrograMin2,
*
resident MinutoMinutoTemp2;

drop table MinutoMinutoTemp2;

//Salida del Script;

// creación del catálogo de los Targets

TargetTemp:
load Distinct
Target
resident Programacion;
Concatenate
load
Target
resident MinutoMinutoTemp3;

Target:
load
AutoNumberHash128(Target) as IdTarget,
Target
resident TargetTemp;
drop Table TargetTemp;
//-----TarjetPersonas

GeneroPersonas:
LOAD
IdTarget,
Target as TargetGeneroPersonas
Resident Target
Where IdTarget = 13 or IdTarget = 15;
STORE GeneroPersonas into [..\QVD\GeneroPersonas.qvd](qvd);
DROP Table GeneroPersonas;

```

TvPagaPersonas:

LOAD

IdTarget,

Target as TargetTvPagaPersonas

Resident Target

Where IdTarget = 28

or IdTarget = 34;

STORE TvPagaPersonas into [..\QVD\TvPagaPersonas.qvd](qvd);

DROP Table TvPagaPersonas;

NSEPersonas:

LOAD

IdTarget,

Target as TargetNSEPersonas

Resident Target

Where IdTarget = 7

or IdTarget = 22

or IdTarget = 12

or IdTarget = 17;

STORE NSEPersonas into [..\QVD\NSEPersonas.qvd](qvd);

DROP Table NSEPersonas;

EdadPersonas:

LOAD

IdTarget,

Target as TargetEdadPersonas

Resident Target

Where IdTarget = 3

or IdTarget = 4

or IdTarget = 5

or IdTarget = 18

or IdTarget = 23

or IdTarget = 33;

STORE EdadPersonas into [..\QVD\EdadPersonas.qvd](qvd);

DROP Table EdadPersonas;

ABCPersonas:

LOAD

IdTarget,

Target as TargetABCPersonas

Resident Target

Where IdTarget = 19

or IdTarget = 21

or IdTarget = 26

or IdTarget = 27

or IdTarget = 35;

STORE ABCPersonas into [..\QVD\ABCPersonas.qvd](qvd);

DROP Table ABCPersonas;

DEPersonas:

LOAD

IdTarget,

Target as TargetDEPersonas

```
Resident Target
Where IdTarget = 6
or IdTarget = 14
or IdTarget = 25
or IdTarget = 30
or IdTarget = 32;
STORE DEPersonas into [..\QVD\DEPersonas.qvd](qvd);
DROP Table DEPersonas;
//-----FinPersonas
//-----TargetsHogares

TvPagaHogares:
LOAD
IdTarget,
Target as TargetTvPagaHogares
Resident Target
Where IdTarget = 2
or IdTarget = 20;
STORE TvPagaHogares into [..\QVD\TvPagaHogares.qvd](qvd);
DROP Table TvPagaHogares;

NSEHogares:
LOAD
IdTarget,
Target as TargetNSEHogares
Resident Target
Where IdTarget = 8
or IdTarget = 11
or IdTarget = 16
or IdTarget = 24;
STORE NSEHogares into [..\QVD\NSEHogares.qvd](qvd);
DROP Table NSEHogares;

ABCHogares:
LOAD
IdTarget,
Target as TargetABCHogares
Resident Target
Where IdTarget = 29;
STORE ABCHogares into [..\QVD\ABCHogares.qvd](qvd);
DROP Table ABCHogares;

DEHogares:
LOAD
IdTarget,
Target as TargetDEHogares
Resident Target
Where IdTarget = 10;
STORE DEHogares into [..\QVD\DEHogares.qvd](qvd);
DROP Table DEHogares;
//-----FinTargetsHogares
store Target into [..\QVD\Target.qvd](qvd);
```

drop table Target;

VariableAudienciaTemp:

load Distinct

Variable

resident Programacion;

Concatenate

load

Variable

resident MinutoMinutoTemp3;

VariableAudiencia:

load

AutoNumberHash128(Variable) as IdVariable,

Variable

resident VariableAudienciaTemp;

drop Table VariableAudienciaTemp;

store VariableAudiencia into [..\QVD\VariableAudiencia.qvd](qvd);

drop Table VariableAudiencia;

DatesTable:

LOAD * INLINE [

Start, End

0:00, 23:59

1:00, 23:59

2:00, 23:59

3:00, 23:59

4:00, 23:59

5:00, 23:59

6:00, 23:59

7:00, 23:59

8:00, 23:59

9:00, 23:59

10:00, 23:59

11:00, 23:59

12:00, 23:59

13:00, 23:59

14:00, 23:59

15:00, 23:59

16:00, 23:59

17:00, 23:59

18:00, 23:59

19:00, 23:59

20:00, 23:59

21:00, 23:59

22:00, 23:59

23:00, 23:59

];

DatesTable2:

LOAD * INLINE [

Start2, End2

0:01, 1:00

0:01, 2:00

0:01, 3:00

0:01, 4:00

0:01, 5:00

0:01, 6:00

0:01, 7:00

0:01, 8:00

0:01, 9:00

0:01, 10:00

0:01, 11:00

0:01, 12:00

0:01, 13:00

0:01, 14:00

0:01, 15:00

0:01, 16:00

0:01, 17:00

0:01, 18:00

0:01, 19:00

0:01, 20:00

0:01, 21:00

0:01, 22:00

0:01, 23:00

0:01, 24:00

];

ProgramacionH :

load

Dominio,

IdCanal,

Date,

Target,

Variable,

PrgTiempoInicio,

PrgTiempoFin,

Description as Programa,

PrgValorVariable

resident Programacion;

ProgramacionH2:

load distinct

(upper(Dominio) & IdCanal & Date & Target & Variable & PrgTiempoInicio & PrgTiempoFin) as

keyProgramacion,

Dominio as DominioR,

IdCanal as IdCanalTrans,

Date as FechaTrans,

Date(Date,'DD/MM/YYYY')as DateF,

AutoNumberHash128(Target) as IdTarget,

```

AutoNumberHash128(Variable) as IdVariable,
AutoNumberHash256(Programa) as IdPrograma,
PrgTiempoInicio,
time(PrgTiempoInicio,'hh:mm:ss') as PrgTiempoInicioF,
PrgTiempoFin,
time(PrgTiempoFin,'hh:mm:ss') as PrgTiempoFinF,
Programa,
PrgValorVariable,
(IdCanal & '_' & Date & '_' & PrgTiempoInicio & '_' & PrgTiempoFin) as %IdIngresos

```

```

resident ProgramacionH;
drop table ProgramacionH;

```

```

ProgramacionH3:
INTERVALMATCH (PrgTiempoInicioF) LEFT JOIN LOAD Start,
    End
RESIDENT DatesTable;

```

```

LEFT JOIN (ProgramacionH2) LOAD *
RESIDENT DatesTable;

```

```

ProgramacionH4:
INTERVALMATCH (PrgTiempoInicioF) LEFT JOIN LOAD Start2,
    End2
RESIDENT DatesTable2;

```

```

LEFT JOIN (ProgramacionH2) LOAD *
RESIDENT DatesTable2;

```

```

store ProgramacionH2 into [..\QVD\ProgramacionH2.qvd](qvd);

```

```

Programa2:

```

```

LOAD keyProgramacion as keyProgramacion2,
    IdTarget,
    IdVariable,
    Programa as Programa2,
    PrgValorVariable as PrgValorVariable2,
    DominioR as DominioR2
Resident ProgramacionH2;

```

```

store Programa2 into [..\QVD\Programa2.qvd](qvd);

```

```

drop tables ProgramacionH2,Programa2;

```

```

// genera intervalos de minutos por cada programa

```

```

drop fields Dominio,IdCanal,Date,Target,Variable,PrgTiempoInicio,PrgTiempoFin from Programacion;
MinutoMinutoTemp4:

```

```

load
keyPrograMin,

```

```
MinutoInicio,
MinValorVariable
resident MinutoMinutoTemp3;
INNER JOIN
IntervalMatch ( MinutoInicio,keyPrograMin )
LOAD
  PrgTiempoInicioT,
  PrgTiempoFinT,
  keyPrograMin
resident Programacion;
drop table Programacion;
drop table MinutoMinutoTemp3;

// genera llave para minuto
MinutoMinuto:
load
MinutoInicio,
MinValorVariable,
keyPrograMin & PrgTiempoInicioT & PrgTiempoFinT as keyProgramacion
resident MinutoMinutoTemp4;

store MinutoMinuto into [..\QVD\MinutoMinuto.qvd](qvd);

MinutoMinuto2:
LOAD MinutoInicio as MinutoInicio2,
  MinValorVariable as MinValorVariable2,
  keyProgramacion as keyProgramacion2
resident MinutoMinuto;

store MinutoMinuto2 into [..\QVD\MinutoMinuto2.qvd](qvd);
drop tables MinutoMinuto, MinutoMinutoTemp4, MinutoMinuto2;

drop table Canal;

EXIT Script
// Termina el código
```

ANEXO C

Acta de Cierre de Proyecto

Sistema de apoyo en la planeación de programación en
televisoras mexicanas

Terminación del sistema

Fecha: 25/septiembre/2015

Tabla de contenido

<i>Información del proyecto</i>	136
Datos	136
<i>Razón de cierre</i>	136
<i>Aceptación de los entregables</i>	136
<i>Aprobaciones</i>	137

Información del proyecto

Datos

Empresa / Organización	
Proyecto	
Fecha	
Cliente	
Patrocinador principal	
Gerentes de Proyecto	

Razón de cierre

Por medio de la presente, se da cierre formal al proyecto debido a que se cumplió formal y oportunamente con la entrega de todos los productos de conformidad con los requerimientos del cliente.

Aceptación de los entregables

A continuación se establecen los entregables de proyecto:

Entregable	Aceptación (Si o No)	Observaciones
Bienvenido		
Análisis de canales		
Análisis de la audiencia		
Análisis de un programa		

Para cada entregable aceptado, se da por entendido que:

- El entregable ha cumplido los criterios de aceptación establecidos en la documentación de requerimientos y definición de alcance.
- Se ha verificado que los entregables cumplen los requerimientos.
- Se ha validado el cumplimiento de los requerimientos funcionales definidos.
- Se ha realizado la transferencia de conocimientos.
- Se ha concluido el curso de capacitación que se definió necesario.
- Se ha entregado la documentación técnica.

Aprobaciones

Nombre	Fecha	Firma

Bibliografía:

Roger S. Pressman. INGENIERÍA DEL SOFTWARE: UN ENFOQUE PRÁCTICO.
Sexta edición. Editorial: MC Graw Hill.

Thomas M. Connolly, Carolyn E. Begg. SISTEMAS DE BASES DE DATOS: Un enfoque práctico para diseño, implementación y gestión. Cuarta edición. Editorial: PEARSON Addison Wesley.

Referencias Electrónicas.

Liga: https://www.nielseniboep.com.mx/uploads/TAM_1.pdf

Tema: Guía para el uso de datos de audiencia.

Autor: Josep Lluís Cano.

Fecha de consulta: 13 Enero del 2015.

Liga: <https://www.iboepagb.com.mx/quienes/asociaciones.php>

http://www.amai.org/que_hacemos.php

<http://www.iabmexico.com/Mediatoca>

<http://www.peoplemeter.cl/timeiboep/grupoiboep.asp>

<http://www.timeiboep.cl/timeiboep/rating.asp>

Tema: ¿Cómo se mide el rating en México?

Autor: Varios.

Fecha de consulta: 30 enero del 2015.

Liga: http://www.softdelsur.com/ORACLE_BI_Inteligencia_de_Negocios.PDF

Tema: ORACLE_BI_Inteligencia_de_Negocios.

Autor: ORACLE.

Fecha de consulta: 15 Febrero del 2015.

Liga: http://itemsweb.esade.edu/biblioteca/archivo/Business_Intelligence_competir_con_informacion.pdf

Tema: Business_Intelligence_competir_con_informacion.

Autor: Josep Lluís Cano.

Fecha de consulta: 28 Febrero del 2015.