UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

Inteligencia de Negocios Aplicada en la Industria Farmacéutica

INFORME DE ACTIVIDADES PROFESIONALES

Que para obtener el título de

Ingeniero en Computación

P R E S E N T A

Carlos López Roa

ASESORA DE INFORME

Dra. María del Pilar Ángeles

Ciudad Universitaria, Cd. Mx., 2019

Contenido

1. Introducción 1
1.1 Objetivo 1
1.2 Antecedentes 1
1.3 Trayectoria Profesional 1
1.3.1 Analista de Estadística 1
1.3.2 Analista de Servicio al Cliente 2
1.3.3 Analista de Sistemas de Negocio 3
2. Descripción de la empresa 4
2.1 La empresa 4
2.2 Organigrama 5
3. Conocimientos previos 5
3.1 Inteligencia de Negocios 6
3.2 Data Warehouse 8
3.3 Modelo Dimensional 9
3.4 Extracción, Transformación y Carga 10
4. Participación profesional 11
4.1 Como Analista de Estadística 11
4.1.1 Elaboración de folios 11
4.1.1.1 Antecedentes 11
- Recepción de la solicitud y asignación del folio 13

4.1.1.2 Metodología de resolución 14
- Elaboración del folio 14
- Pruebas realizadas 15
- Problemas presentados 19

4.1.1.3 Resultados obtenidos 20
4.1.2 Análisis de ventas 21
4.1.2.1 Antecedentes 21
4.1.2.2 Metodología de resolución 22
- Extracción de los datos 22
- Preparación de los datos 22
- Tratamiento de los datos 23
- Interpretación de la información 26

4.1.2.3 Resultados obtenidos 28
4.1.3 Optimización 29
4.1.3.1 Antecedentes 29
4.1.3.2 Metodología de resolución 30
- Pruebas realizadas 30

4.1.3.3 Resultados obtenidos 32
4.2 Como analista de Servicio al Cliente 33
4.2.1 Explotación de Bases de Datos 33
4.2.1.1 Antecedentes 33
4.2.1.2 Metodología de resolución 34
- Consulta caso real 35
4.2.1.3 Resultados obtenidos 36
4.2.2 Automatización para corrección de inconsistencias 37
4.2.2.1 Antecedentes 37
4.2.2.2 Metodología de resolución 37
- Pruebas realizadas 37
4.2.2.3 Resultados obtenidos 43
4.2.3 Validación de venta interna 43
4.2.3.1 Antecedentes 43
4.2.3.2 Metodología de resolución 43
4.2.3.3 Resultados obtenidos 44
4.3 Como analista de Sistemas de Negocio 45
4.3.1 Diseño de procesos ETL, construcción de cubos y administración de herramientas BI 45
4.3.1.1 Antecedentes 45
4.3.1.2 Metodología de resolución 46
- Extracción 46
  - Pruebas realizadas 47
- Transformación 48
- Carga 53
4.3.1.3 Resultados obtenidos 54
7. Conclusiones 55
8. Referencias 55

Índice de Tablas

Tabla 1: Funciones a Desarrollar 4
Tabla 2: OLAP vs OLTP 9
Tabla 3: Conocimientos Previos 10

Índice de Ilustraciones

Ilustración 1: Refinería de Datos 6
Ilustración 2: Arquitectura de Inteligencia de Negocios 8
Ilustración 3: Caso Real - Solicitud de Información 14
Ilustración 4: Caso Real - Primera parte del Layout 14
Ilustración 5: Caso Real - Segunda Parte del Layout 15
Ilustración 6: Modelo Dimensional del Data Warehouse 15
Ilustración 7: Preparación de los datos 22
Ilustración 8: Ventas Laboratorio 1 26
Ilustración 9: Ventas Laboratorio 2 27
Ilustración 10: BoxPlot 27
Ilustración 11: Plantilla de Dashboard en Power BI 45
Ilustración 12: Carga de Catálogos 47
Ilustración 13: Data Cruda 48
Ilustración 14: Transformación a Tabla de Hechos con SSIS 50
Ilustración 15: Mapeo de Columnas 51
Ilustración 16: Joins en SSIS 51
Ilustración 17: T-SQL en SSIS 52
Ilustración 18: Sistema ETL en SSIS 53

Índice de Figuras

Figura 1: Organigrama 5
Figura 2: Áreas Involucradas en Estadística 12
Figura 3: Proceso General para Elaborar un Folio 12
Figura 4: Estructura de la Solicitud de Información 13
Figura 5: Cubo de Información 19
Figura 6: Fragmentación de la Tabla 20
Figura 7: Tipos de Venta 21
Figura 8: Rango Intercuartílico 25
Figura 9: Detección de Anomalías Categorizadas 28
Figura 10: Proceso de Análisis de Ventas 29
Figura 11: Diagrama de Flujo de SP_FOLIOS 31
Figura 12: Estructura del Área de Servicio al Cliente 34
Figura 13: Flujo de Datos y Trabajo 35
Figura 14: Diagrama de Flujo SP_VENTA_VS 40
Figura 15: Proceso de Validación de Venta Interna 44
Figura 16: Flujo de Ejecutables 50
Figura 17: Sistema ETL 54

Índice de Códigos

Código 1: Maestro 16
Código 2: Mercado 17
Código 3: Cruce Mensual 18
Código 4: Generación de BoxPlot en Lenguaje R 24
Código 5: SP_FOLIOS 32
Código 6: Script Consulta a Cliente 36
Código 7: SP_VENTA_VS 42
Código 8: Bulk Insert 47
Código 9: SP_DERIVA_COLUMNAS 49
Código 10: Sentencia CONNECT_BD_EXEC_SP en SQLCMD 50

1. Introducción

En el presente informe describiré las principales actividades y funciones que realicé en una empresa del sector farmacéutico y de la salud en el puesto de *Analista de Datos* en las áreas de **Estadística** y **Servicio al Cliente**, y como *Analista de Sistemas de Negocio* en el área de **Tecnología y Aplicaciones,** así como también detallaré los procesos necesarios, los conocimientos previos y las habilidades técnicas pertenecientes al área de Ingeniería en Computación que me permitieron llevar a cabo dichas actividades satisfactoriamente y los impactos positivos que tuvieron en beneficio de dichas áreas.

1.1 Objetivo

El objetivo principal es demostrar, mediante este informe, el correcto uso de los conocimientos, habilidades y técnicas adquiridos del plan de estudios de la licenciatura en ingeniería en computación, así como de los adquiridos a través de toda mi trayectoria escolar, y su correcta aplicación en el campo laboral dentro de proyectos reales, demostrando así, que cuento con los requisitos demandados actualmente por la industria de tecnologías de la información en México.

1.2 Antecedentes

La cantidad de información que se genera actualmente ha crecido de forma increíblemente grande en los últimos años gracias al desarrollo y uso de nuevas tecnologías de información y formas de comunicación.

Las empresas, sin importar el giro al que pertenezcan, de igual forma generan enormes cantidades de datos los cuales necesitan ser tratados y explotados para su posterior transformación en información que pueda ser útil para elaborar planes y estrategias que incrementen el valor de su negocio.

Existen organizaciones encargadas de ofrecer tales servicios mencionados anteriormente para que las compañías que las contraten puedan elaborar planes de estrategia empresarial. Todo esto es conocido como Inteligencia de Negocios.

Una vez mencionado el contexto, explicaré de forma breve mi trayectoria laboral dentro de la empresa, destacando mis funciones principales.

1.3 Trayectoria Profesional

A continuación, mencionaré los puestos de trabajo que ocupé dentro de la empresa, la duración, y las actividades destacadas que realicé en cada uno.

1.3.1 Analista de Estadística

Fungí como parte del equipo de estadística, en el rol de becario, a partir de junio de 2016 hasta agosto de 2017, posteriormente me integré como analista de datos en el periodo comprendido de octubre de 2017 a marzo de 2018.

El área de estadística es la encargada, principalmente, de realizar el cálculo de factores de proyección que se aplican a la data mensual, de desarrollar la metodología para el cálculo y asignación de precios a los productos, de realizar pronósticos de la data que entregan a la empresa cada uno de los proveedores y de darle soporte al área de consultoría.

Como analista de estadística realicé, entre otras, las siguientes actividades:

- Validé datos provenientes de distintas fuentes.
- Realicé scripts en SQL para la extracción y lectura de los datos.
- Elaboré catálogos que facilitaron el cruce de diversas fuentes de datos.
- Limpié datos sucios que impactaban en la comprensión de la información.
- Programé procedimientos almacenados en SQL que redujeron el tiempo de elaboración de reportes para los distintos departamentos quienes solicitaban nuestro apoyo.
- Programé procedimientos almacenados en SQL que optimizaron el manejo de datos en proyectos de la misma área.
- Llevé a cabo tareas de administración y mantenimiento a las bases de datos, pues cargaba los respaldos mensuales, liberaba espacio en memoria y otorgaba permisos a distintos usuarios.
- Redacté manuales de SQL para la elaboración de folios.
- Redacté manuales que definen los procesos necesarios para la realización de folios.
- Realicé análisis de las ventas de entrada contra las ventas de salida para validar la calidad de la información.
- Programé scripts en lenguaje R que ejecutaban diferentes cálculos estadísticos para la comprensión de los datos.
- Programé scripts en lenguaje R que creaban gráficos para la visualización y fácil entendimiento de la información.

1.3.2 Analista de Servicio al Cliente

En marzo de 2018 me uní al equipo de servicio al cliente, igual que en el puesto anterior, fui analista de datos.

El área es la encargada de monitorear las ventas de los productos de cada laboratorio o cliente, las cuales son visibles a través de distintas plataformas de BI. Además, el área se encarga de liberar la data mensual a todos los clientes después de varios procesos de validación, así como también de dar soporte y asesoría concerniente a la comprensión del negocio referente a la información liberada a cada cliente.

Como analista de datos de servicio al cliente realicé las siguientes labores:

- Realicé consultas en SQL de distintas bases de datos que me eran solicitadas para la detección de anomalías o resolución de dudas respecto a la comprensión de la información.
- Dado mi acceso total y directo a las bases de datos, acudí como apoyo para explotar las bases de datos con el ejecutivo de cuenta a reuniones directamente con el cliente para la resolución de dudas o anomalías que éste mismo necesitaba resolver al instante.
- Validé la data mensual que era recibida por parte del área de producción con el fin de liberarla a cliente sin error alguno.
- Programé procedimientos almacenados en SQL para la corrección de errores derivados de procesos ejecutados por diversas áreas.
- Realicé validaciones de venta interna de potenciales clientes interesados en contratar alguna auditoria; estas validaciones eran presentadas en herramientas como Power BI.

1.3.3 Analista de Sistemas de Negocio

A partir de marzo 2019 me integré al equipo de Tecnología y Aplicaciones como analista de sistemas de negocio, cargo que desempeño actualmente.

Esta área se encarga del diseño y construcción del data warehouse, así como de la construcción y posterior carga de los cubos de información para cada cliente a las herramientas de BI, al mantenimiento y administración de dichas herramientas y plataformas, y al soporte continuo a los clientes para el correcto uso de las mismas.

Como analista de sistemas de negocio llevé a cabo lo siguiente:

- Diseñé procesos de integración de datos provenientes de distintas fuentes.
- Construí cubos de información basados en el modelo estrella.
- Generé la estructura de las tablas de hechos y las tablas catálogo, dentro de la herramienta de BI, para posteriormente ser alimentadas.
- Validé y liberé los cubos de información a los ejecutivos de cuenta y a los clientes mismos.
- Administré dicha herramienta de BI, otorgué permisos a usuario y di mantenimiento a los modelos de datos previamente cargados.
- Atendí peticiones de usuarios, pertenecientes a otras áreas dentro de la empresa, referentes al uso y desempeño de la herramienta.
- Diseñé procesos de extracción, transformación y carga para la elaboración de reportes customizados.
- Acudí a reuniones directamente con clientes para presentar la herramienta BI.
- Acudí a reuniones directamente con clientes, en compañía de miembros de otras áreas, para planificar la entrega de reportes personalizados, estableciendo la información a entregar y fijando costos.

En los tres puntos anteriores, describí las áreas en las que laboré, así como mis principales funciones. En el capítulo número cuatro desarrollaré lo que será el eje principal de este informe, ya que explicaré a mayor detalle las actividades más significativas y de mayor importancia que me fueron asignadas en las áreas en cuestión, mencionando el origen de dichas actividades, su desarrollo y los impactos obtenidos de su elaboración.

Puesto	Actividades principales a desarrollar
Analista de Estadística	✓ Elaboración de folios ✓ Análisis de ventas ✓ Optimización
Analista de Servicio al Cliente	✓ Explotación de bases de datos ✓ Automatización para corrección de inconsistencias ✓ Validación de venta interna

Analista de Sistemas de Negocio	✓ Diseño de procesos ETL, construcción de cubos y administración de herramienta BI

Tabla 1: Funciones a Desarrollar

En la Tabla 1 muestro las funciones más significativas que desempeñé en cada puesto que ocupé dentro de la empresa, las cuales desarrollaré a profundidad en este informe.

En este primer capítulo establecí un punto de partida para entender el tema principal que desarrollaré a lo largo de este informe y mencioné las funciones que realicé en la empresa. En el capítulo siguiente, hablaré brevemente de la empresa y su organización interna.

2. Descripción de la empresa

2.1 La empresa

Por motivos de privacidad a la empresa, a lo largo de este informe me referiré a ella como QH Company, la cual, es una empresa líder en el país dentro del sector farmacéutico y de la salud que utiliza la ciencia de datos y la inteligencia de negocios con la finalidad de ofrecer servicios y soluciones referentes a la explotación y análisis de información a un amplio portafolio de clientes para que ellos, a su vez, puedan tomar decisiones que impacten positivamente a su negocio.

QH Company es capaz de ofrecer soluciones gracias a sus numerosas y vastas fuentes de información, las cuales, a través del trabajo en conjunto de procesamiento de datos de sus departamentos, son transformadas en información de alto valor para sus clientes.

2.2 Organigrama

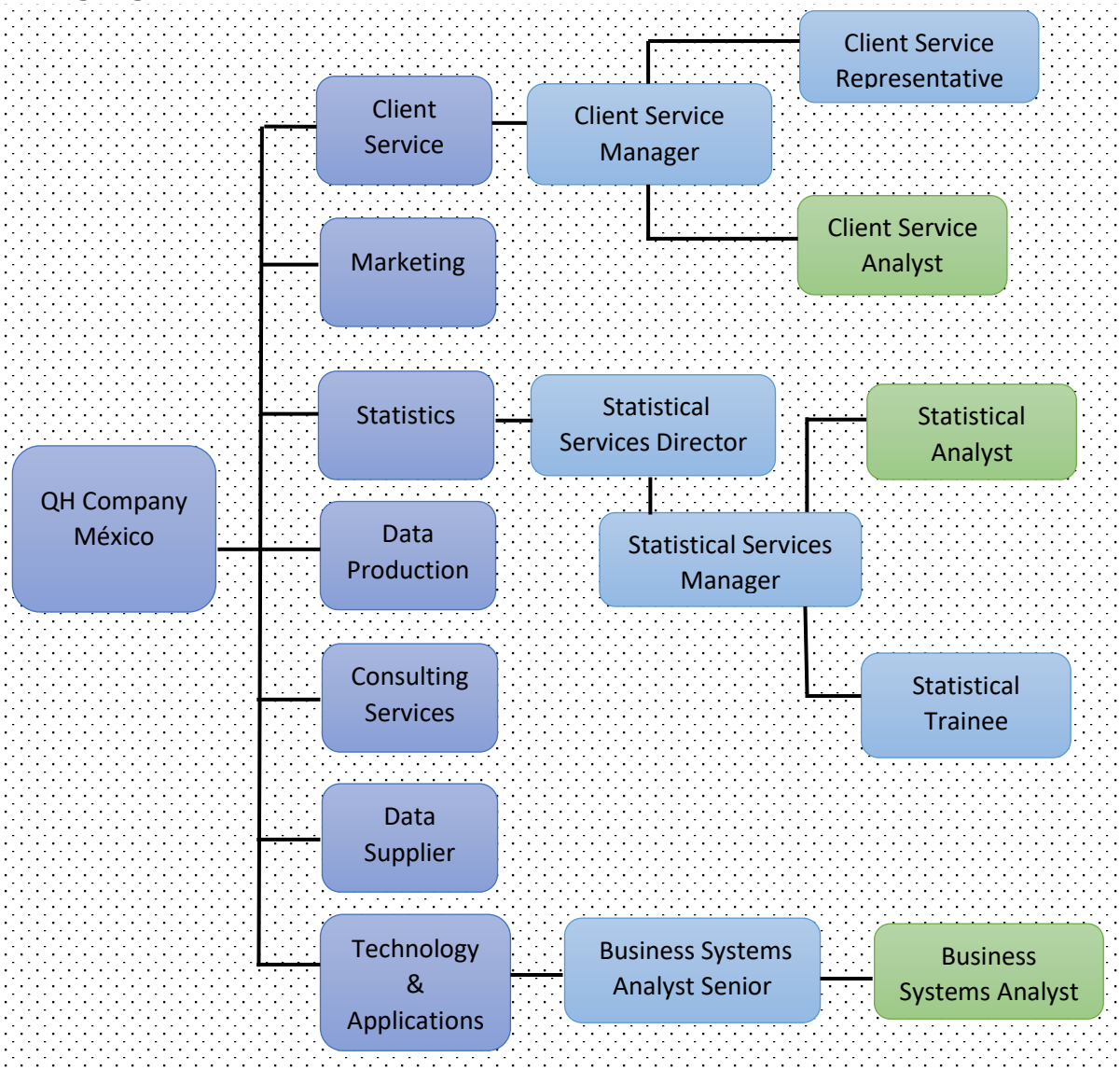


Figura 1: Organigrama

En la Figura 1 se muestran algunos departamentos de la empresa. Aquellos donde desglosé la jerarquía a mayor detalle son las áreas a las que pertenecí; los puestos que ocupé están remarcados con diferente color.

En esta sección expliqué el sector al que pertenece la empresa y de manera general mencioné qué es a lo que se dedica, así como también mostré la organización dentro de las áreas en las que laboré.

Antes de continuar, es conveniente remarcar los conocimientos obtenidos del plan de estudios y las bases teóricas que me sirvieron para desempeñar los cargos ya mencionados.

3. Conocimientos previos

Para poder desempeñar exitosamente las funciones de las áreas en las que trabajé fue necesario contar con conocimientos antecedentes que fui adquiriendo a lo largo de la carrera universitaria. Estos conocimientos me dieron los cimientos y las bases suficientes para enfrentar y solucionar los problemas de los puestos en el que laboré.

Es importante mencionar las bases teóricas de los temas ejes desarrollados en este informe, para así facilitar la comprensión del mismo. Cabe mencionar, que tales temas

fueron los mínimos necesarios para poder comprender la problemática de las labores para las que fui contratado y, por ende, para encontrar soluciones a problemas derivados de dichas labores.

3.1 Inteligencia de Negocios

La Inteligencia de Negocios o *Business Intelligence (BI)* es la combinación de diversas tecnologías, herramientas, metodologías y procesos que permiten transformar datos – los cuales se encuentran previamente almacenados en un repositorio – en información; esta información, a su vez, es transformada en conocimiento, y este conocimiento obtenido es dirigido hacia la toma de decisiones para la elaboración de planes o estrategias comerciales con la finalidad de generar impactos positivos en el negocio. [1]

Para tener un mejor entendimiento acerca de lo que es la inteligencia de negocios, podemos pensar en este término como si se tratase de una refinería de datos. En esta, se extraen los datos provenientes de distintos sistemas operacionales y son cargados a un repositorio de datos, después son analizados con diferentes herramientas con el fin de transformarlos en información y, posteriormente, dicha información se convierte en acciones a llevar a cabo en los planes y estrategias comerciales. [2]

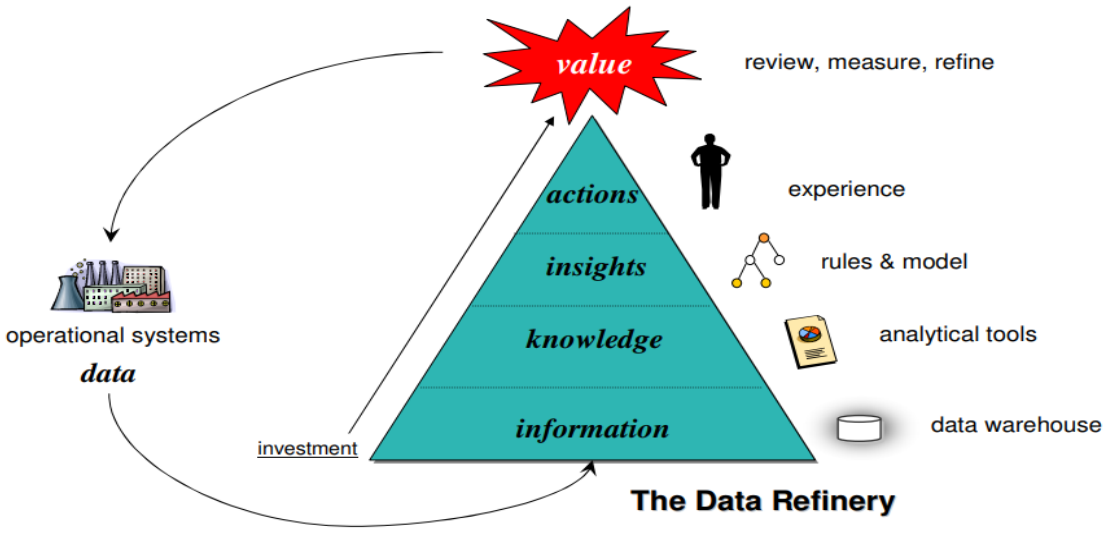


Ilustración 1: Refinería de Datos

La Ilustración 1 [2] muestra la refinería de datos, la cual es una analogía para comprender mejor el término de Inteligencia de Negocios.

Arquitectura de Inteligencia de Negocios

Previamente a explicar la arquitectura típica de Inteligencia de Negocios, se debe hacer énfasis en la importancia de conocer y tener definidos los objetivos estratégicos del área del negocio para determinar qué es lo que realmente se requiere medir, ya que, sin esto, no se tendrá un punto de partida.
Una vez que lo anterior se encuentra establecido, se realiza un trabajo de análisis con la colaboración de los usuarios experimentados, esto permitirá identificar las fuentes y proveniencia de los datos necesarios. [3]

La arquitectura comienza por la identificación de los datos requeridos, los cuales pueden provenir de múltiples fuentes de información, tales como archivos planos (.txt),

hojas de cálculo (.xlsx), archivos de bases de datos (.dbf), archivos XML (.xml), o bien, pueden provenir de otras bases de datos en otros sistemas productivos.

Inmediatamente después se aplican procesos de extracción, transformación y carga (ETL) desde dichas fuentes a un repositorio, almacén de datos o *data warehouse* (DWH).

Durante los procesos ETL es donde se definen, de las fuentes, los campos que se van a utilizar, si necesitan algún tipo de modificación y/o transformación y se define la ubicación final o destino de estos datos, a este proceso se le conoce como *mapping*. [1]

La arquitectura continúa una vez que los datos se encuentran cargados y transformados correctamente en este almacén de datos corporativo, son representados visualmente en modelos multidimensionales formados por tablas de hechos y tablas de dimensiones. [1]

Dicho repositorio sirve como base para la construcción de *datamarts*, los cuales son modelos de datos, pertenecientes al mismo data warehouse, pero que se caracterizan por tener una estructura enfocada solamente en un tema del negocio, ya sea mediante bases de datos transaccionales (OLTP) o bases de datos analíticas (OLAP) [4]. Dichos conceptos son explicados más adelante.

Los datos almacenados en el repositorio o en cada datamart se pueden explotar usando directamente SQL aplicando las agregaciones necesarias para obtener análisis descriptivos, o bien, utilizando herramientas comerciales de análisis de datos, capaces de generar análisis tanto descriptivos como predictivos de la información. [4]

La última parte de la arquitectura se enfoca en el diseño de la visualización para mostrar las salidas de información. Los instrumentos más clásicos son los informes (reporting), los modelos de análisis multidimensional (OLAP) y los cuadros de mando (scorecard y dashboard).
La alternativa que se elija estará en función de lo que se desea medir y del usuario que hará uso de la información. Una de estas alternativas no excluye a la otra, son técnicas complementarias en ambos sentidos. [3]

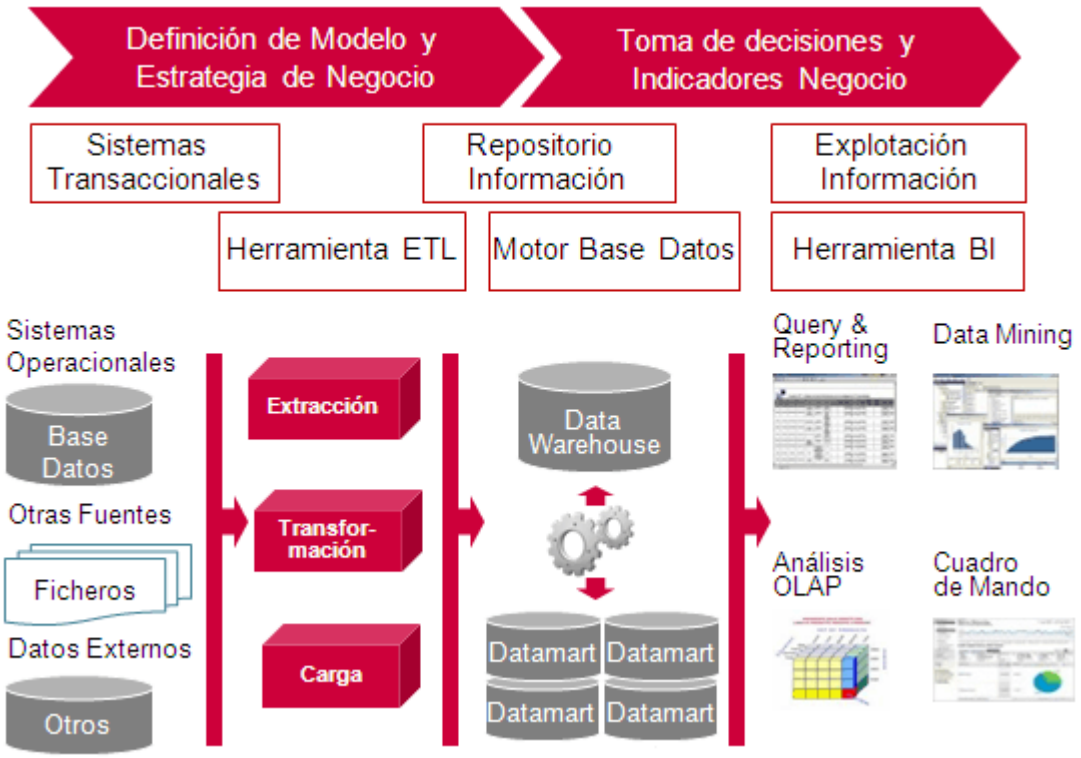


Ilustración 2: Arquitectura de Inteligencia de Negocios

En la Ilustración 2 [4] se presenta la arquitectura típica de un ambiente de inteligencia de negocios. Se muestra desde la etapa donde se definen las fuentes de datos, pasando por los procesos ETL, el repositorio DWH y finaliza con las explotación y visualización de la información.

Dado que los roles y funciones que detallaré en este informe están más enfocadas en la parte referente al almacén de datos, así como de los procesos contiguos a este, es conveniente definir dichos conceptos.

3.2 Data Warehouse

Data Warehouse – traducido literalmente como almacén, repositorio o depósito de datos – es una tecnología para el manejo de la información construido con la finalidad de optimizar el uso y análisis de ésta utilizado por las organizaciones. La función esencial de estas bases de datos empresariales es ser el rol integrador de toda la información resultante de los sistemas transaccionales y brindar una visión integrada de la misma, enfocada hacia la toma de decisiones por parte del personal jerárquico de la empresa. [5]

La ventaja principal de este tipo de bases de datos radica en las estructuras donde se almacena la información, pues utilizan el modelo multidimensional, lo que permite la consulta y el tratamiento jerarquizado de la misma. [6]

Ya que con el modelo multidimensional utilizado en los data warehouse es posible obtener a los datos con alto grado de agregación y desde distintas perspectivas (dimensiones), es mayormente utilizado por personal de niveles ejecutivos [5]

Sistema de bases de datos tradicional	Data Warehouse
Predomina la actualización	Predomina la consulta
La actividad más importante es de tipo operativo (día a día)	La actividad más importante es el análisis y la decisión estratégica
Predomina el proceso puntual	Predomina el proceso masivo
Mayor importancia a la estabilidad	Mayor importancia al dinamismo
Datos en general desagregados	Datos en distintos niveles de detalle y agregación
Importancia del dato actual	Importancia del dato histórico
Importante del tiempo de respuesta de la transacción instantánea	Importancia de la respuesta masiva
Estructura relacional	Visión multidimensional
Usuarios de perfiles medios o bajos	Usuarios de perfiles altos
Explotación de la información relacionada con la operativa de cada aplicación	Explotación de toda la información interna y externa relacionada con el negocio

Tabla 2: OLAP vs OLTP

En la Tabla 2 [6] presento las diferencias entre un sistema de bases de datos tradicional (OLTP) y un almacén de datos (OLAP).

Ahora que el concepto de almacén de datos está definido y, sabiendo que se basa en el modelo dimensional, es necesario definir dicho modelo.

3.3 Modelo Dimensional

Modela las singularidades de los diversos procesos que ocurren dentro de una empresa, dividiéndolos en mediciones y entorno. Las mediciones son en su mayoría, medidas numéricas, y se les denomina hechos. Alrededor de estos hechos existe un entorno que describe bajo qué condiciones se registró este hecho, a estas se les denomina dimensiones y, a diferencia de los hechos que son numéricos, estas son, principalmente, textos descriptivos. [7]

Las medidas se registran en las tablas de hechos, siendo la llave de esta tabla, la combinación de las múltiples llaves foráneas que hacen referencia a las dimensiones que describen la ocurrencia de este hecho, en otras palabras, cada una de las llaves foráneas presentes en la tabla de hechos corresponden con la llave primaria de una dimensión. [7]

Cubos OLAP

Los cubos OLAP, cubos de procesamiento analítico en línea, por su acrónimo en inglés, son estructuras de datos multidimensionales usadas en los sistemas de inteligencia de negocios, cuyo objetivo es agilizar la consulta de grandes cantidades de datos. [8]

A diferencia de los sistemas OLTP – procesamiento de transacciones en línea – en donde la cantidad de datos no es robusta, pero predominan las operaciones de lectura y escritura a la base de datos (inserciones, actualizaciones y borrados), los sistemas OLAP están enfocados en la consulta y análisis de la información en todas las dimensiones posibles.

Los cubos fungen como un elemento clave pues permiten tener información pre agregada con todas las combinaciones posibles desde la perspectiva de cualquier dimensión pudiendo, de esta forma, visualizar las métricas que sean de interés para el usuario. [9]

Las estructuras multidimensionales están compuestas por métricas y dimensiones, estas últimas cuentan con jerarquías y niveles. En la explotación de estos cubos es posible navegar por el mismo e ir al detalle de la información deseado, desde cualquier jerarquía de alguna dimensión y mostrando las métricas con cualquier agregación y/o agrupación. [9]

Básicamente, el cubo OLAP, cuyo nombre proviene de su característica multidimensional, es una base de datos que posee diversas dimensiones. [8]

3.4 Extracción, Transformación y Carga

La abreviación ETL significa extracción, transformación y carga, por su acrónimo en inglés. Un proceso ETL se utiliza para realizar la integración de datos provenientes de múltiples fuentes para la construcción de un almacén de datos. Durante este proceso, los datos se extraen de las distintas fuentes de origen, se realizan las trasformaciones necesarias para llevarlos a un formato adecuado y se cargan a un data warehouse u otro sistema con el fin de ser explotados. [10]

Una vez mencionadas las principales bases teóricas que me sirvieron como punto de partida en mi desempeño laboral es conveniente mencionar aquellas asignaturas del plan de estudios que de igual forma fungieron como antecedentes.

Asignatura	Tema(s) aplicados en mi puesto laboral
Bases de Datos	✓ Lenguaje de consulta estructurado SQL. ✓ Programación de procedimientos almacenados. ✓ Diseño de bases de datos relacionales. ✓ Álgebra relacional.
Bases de Datos Distribuidas	✓ Optimización de consultas. ✓ Fragmentación de tablas.
Depósitos de Datos	✓ Diseño y arquitectura de un almacén de datos.
Temas Selectos de Bases de Datos	✓ Calidad de Datos. ✓ Limpieza de Datos. ✓ Inteligencia de negocios.
Estructuras Discretas	✓ Lógica booleana. ✓ Teoría de conjuntos. ✓ Cálculo de predicados.
Bases de Datos Espaciales	✓ Elaboración de subconsultas.
Computación para Ingenieros	✓ Lógica de programación.
Ingeniería de Software	✓ Metodologías de desarrollo de proyectos. ✓ Documentación de software. ✓ Relaciones laborales.
Redacción y Exposición de Temas de Ingeniería	✓ Redacción del español culto. ✓ Redacción de manuales técnicos.
Desarrollo Empresarial	✓ Estructura de una empresa. ✓ Necesidades del mercado. ✓ Organización del tiempo.

Tabla 3: Conocimientos Previos

En la Tabla 3 listo las asignaturas de la carrera y los temas estudiados en las mismas que he aplicado en el campo laboral.

En este capítulo demostré la relación de mis actividades profesionales mencionadas en el capítulo dos con los antecedentes teóricos adquiridos durante la carrera, además, destaqué aquellas asignaturas relevantes para mi desempeño dentro de la empresa.

En el siguiente capítulo desglosaré las actividades mostradas en la tabla 1 del capítulo 1, con el objetivo de detallar su elaboración, los problemas que surgieron en el camino y las soluciones a estos mismos.

4. Participación profesional

En el presente apartado describiré mi desenvolvimiento profesional explicando a mayor detalle técnico las actividades que mencioné en la tabla 1 del capítulo 1, los problemas al realizarlas y las soluciones encontradas a dichos problemas.

Para llevar una mejor comprensión y organización, primero describiré la función o problema que atendí explicando el panorama o antecedente acerca del origen del problema, inmediatamente después detallaré la metodología que utilicé para su resolución y, en caso de que apliqué, explicaré las pruebas realizadas, finalmente mencionaré los impactos de mi participación en dichas actividades o tareas.

4.1 Como Analista de Estadística

4.1.1 Elaboración de folios

El área de estadística daba soporte al área de consultoría en proyectos internos de la misma, dentro de mis actividades principales elaboré folios solicitados por esta área.
Un folio era un reporte de ventas que solicitaba consultoría y el cual formaba parte de un proyecto que se realizaba a petición de un cliente determinado. Dicho reporte se elaboraba con base en ciertas dimensiones, esto es, la solicitud para un folio era solamente para una gama de productos W con presencia en determinados lugares de venta X bajo ciertos distribuidores Y durante un periodo de tiempo Z.

4.1.1.1 Antecedentes

La razón por la cual estadística daba soporte a consultoría era porque personal de la primera tenía perfiles matemático-ingenieriles, a diferencia del personal de consultoría donde eran relacionados a ciencias de la salud.

Debido a que un solo proyecto necesitaba gran cantidad de información para llevarse a cabo, me pedían varios reportes solicitando datos específicos para los diferentes análisis. Estos reportes eran llamados folios. Por cada solicitud de información requerida se asignaba un folio diferente.

Resumiendo lo anterior, el cliente solicitaba mediante un proyecto a consultoría determinada información con la cual resolverían incógnitas y tomarían decisiones. Para trabajar en el proyecto consultoría me realizaba múltiples solicitudes de análisis de datos y yo, por mi parte, las asignaba a folios y devolvía los reportes pedidos para que consultoría pudiera interpretarlos y completar el proyecto inicial pedido por el cliente.

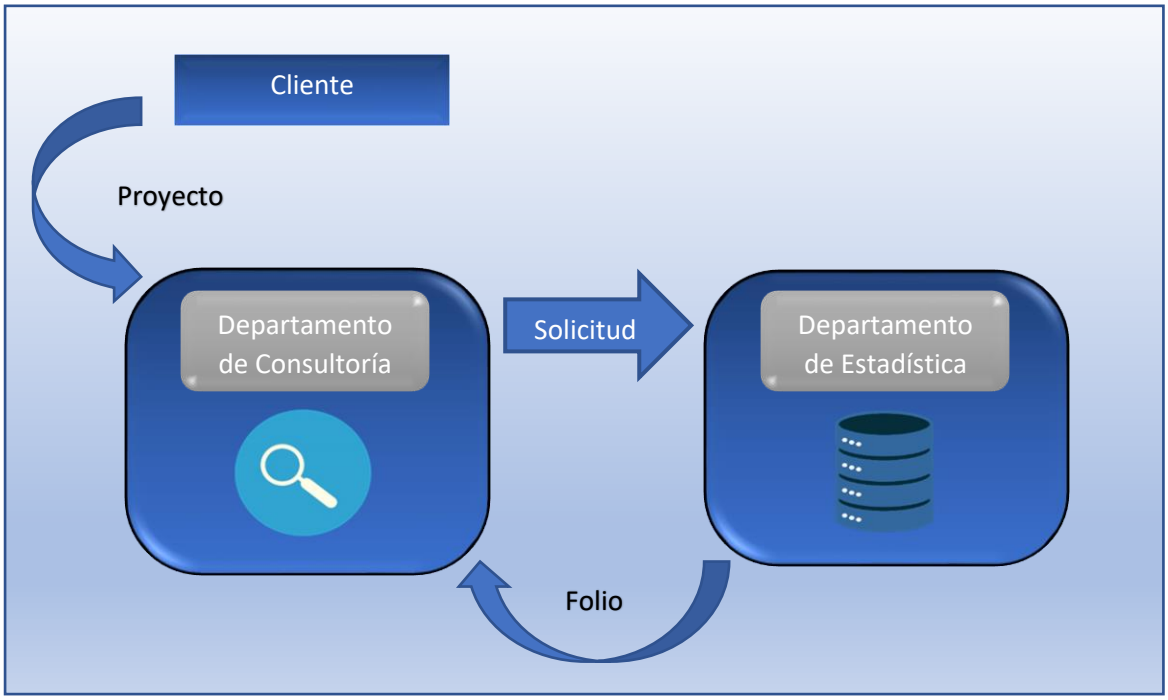


Figura 2: Áreas Involucradas en Estadística

En la Figura 2 represento la relación de las áreas involucradas y el proceso de petición de un folio.
Como ya lo mencioné previamente, el área de estadística, debido a su función, tenía acceso a las bases de datos de las cuales consultoría requería información.

Una vez que se tenía definido un proyecto y se comenzaba a trabajar sobre él, el departamento de consultoría realizaba solicitudes de información a estadística en donde mi función principal fue atender estas solicitudes y asignarlas a un folio.

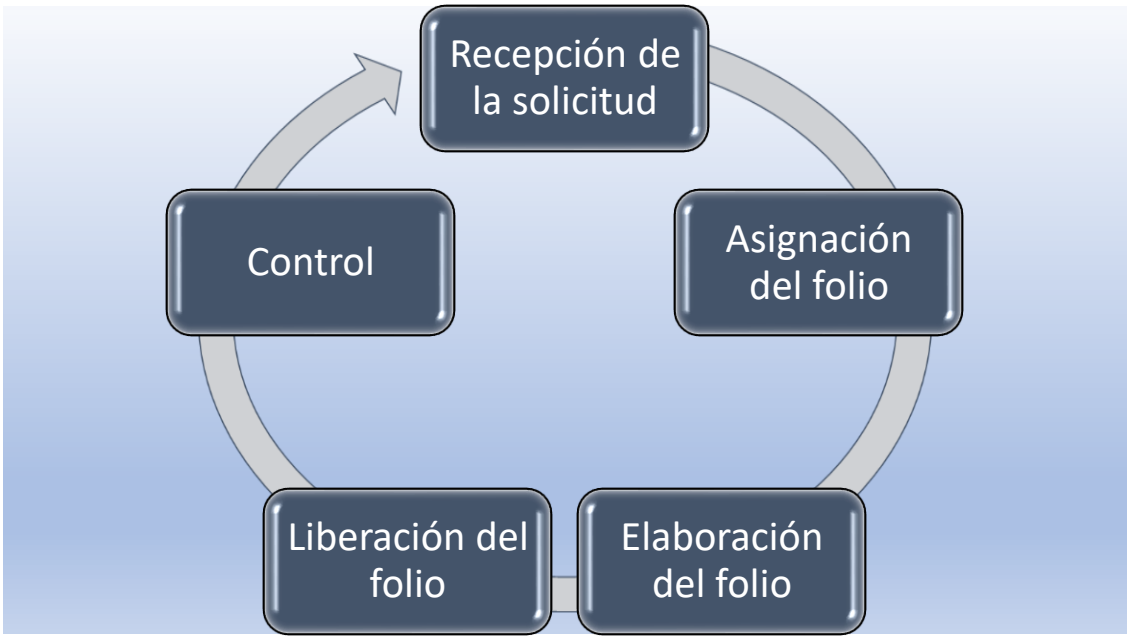


Figura 3: Proceso General para Elaborar un Folio

En la Figura 3 se representan los procesos que se llevaban a cabo para entregar un folio al área de consultoría, desde la recepción de la solicitud hasta la entrega del folio.

- Recepción de la solicitud y asignación del folio

Una vez que la solicitud de información era recibida y asignada a un folio, me encargaba de notificar a ambas áreas que ya me encontraba trabajando sobre él, esto con la finalidad de que consultoría pudiera estimar tiempos y fechas para ir avanzando sobre el proyecto.

Cada solicitud de información estaba dividida principalmente en tres partes: requerimientos, definición de mercado y layout.

- La parte de requerimientos a su vez se encontraba subdividida en: datos generales y características del reporte.

  Los datos generales constaban del nombre del proyecto, el cliente para el cual se estaba realizando, la fecha de envío, la fecha esperada de recepción y los nombres del consultor analista y el consultor a cargo que enviaban la petición.

  En características del reporte se especificaban parte de las dimensiones, es decir, el periodo de tiempo y la periodicidad requerida, la métrica deseada y demás dimensiones para definir el nivel de apertura. Esto último se refería a qué tanto detalle de información el folio debía cubrir, era muy importante prestar atención a esto ya que no estaba permitido dar información muy específica.

- La definición de mercado estaba conformada por conjuntos específicos de productos los cuales podían estar clasificados de diversas formas, por ejemplo, por tipo de componente químico, por forma de aplicación o por presentación.

- El layout contenía el resto de la información que necesitaban y se especificaba, además, cómo la información debía ser presentada, es decir, el formato en el cual ellos deseaban verla para que fuera legible y fácilmente comprendida.

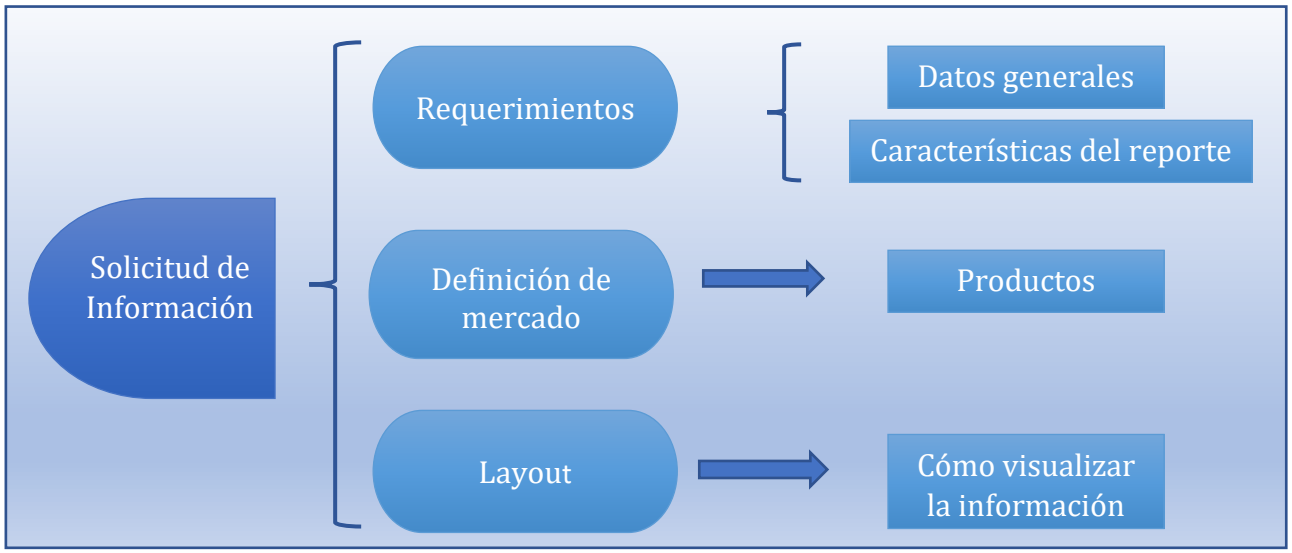


Figura 4: Estructura de la Solicitud de Información

La Figura 4 muestra un esquema de la estructura de las solicitudes de información y su organización.

4.1.1.2 Metodología de resolución

- Elaboración del folio

Para explicar esta parte tomé como ejemplo un caso real de un folio solicitado explicando los pasos a nivel técnico que realicé para su correcta elaboración. Partiré del momento en dónde analicé los requerimientos, definí las dimensiones pedidas y los traduje a código SQL para su extracción. Para proteger la privacidad de la información que maneja la empresa, los datos reales no serán mostrados.

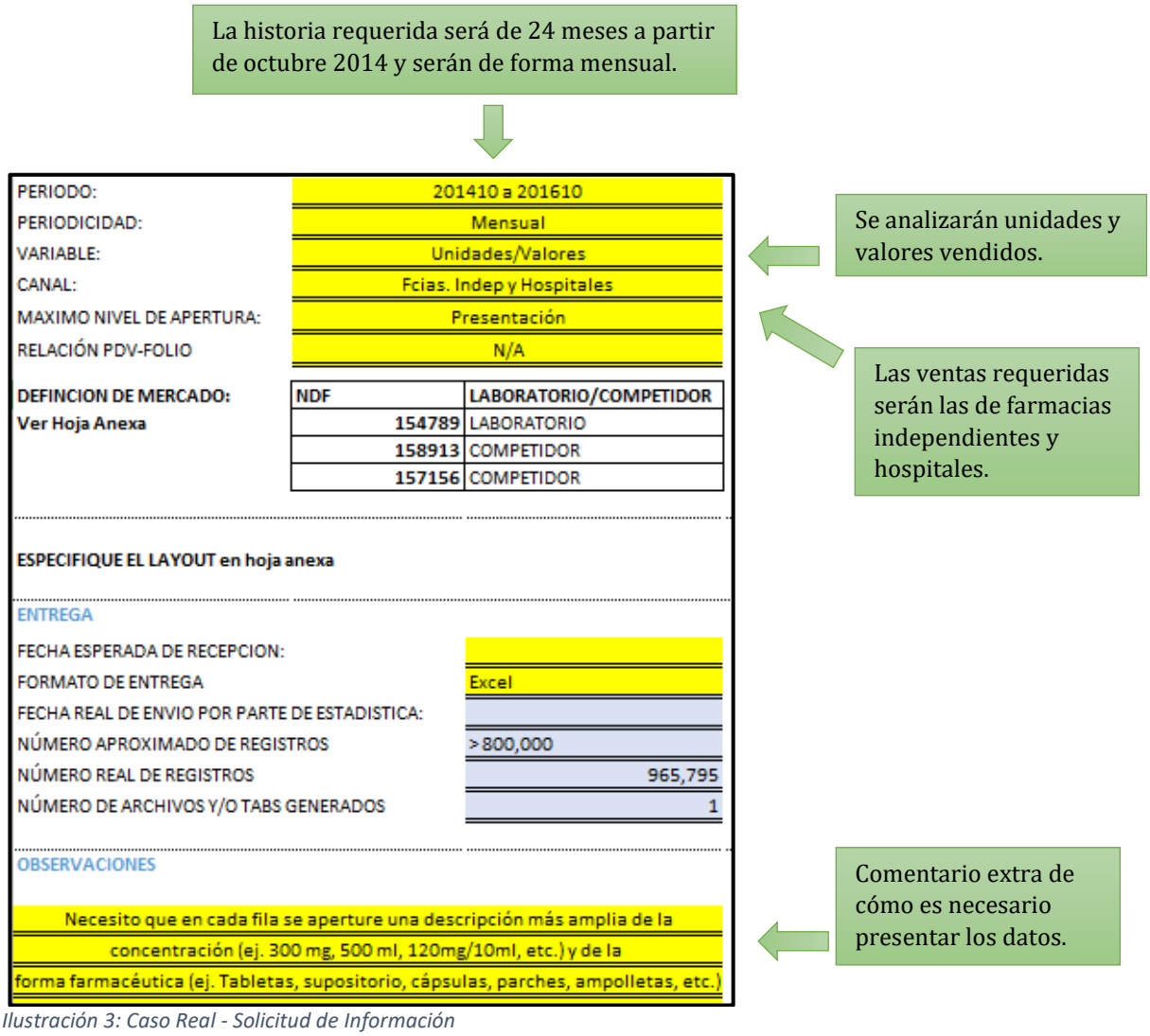


Ilustración 3: Caso Real - Solicitud de Información

En la Ilustración 3 muestro una solicitud que recibí con los datos que necesito para comenzar a trabajarla. En ella se muestran algunas variables que requiero.

code	presentacion	DESCRIP	CONCENTR_SOL	SOLIDO	CONCENTR_LIQ	LIQUIDO	PRESENTACIÓN
123	PRODEJEMPLO_1 CREMA TUBO 1% 24 G x 1	24 G	24	G	-	-	CREMA
456	PRODEJEMPLO_2 SOLN INF 160 MG 120 ML x 1 (/5ML	160 MG 120 ML	160	MG	120	ML	SOLN INF
789	PRODEJEMPLO_3 SUP. ADLT 300 MG x 5	300 MG	300	MG			SUSP. ADLT
101	PRODEJEMPLO_4 GOTAS 100 MG 15 ML x 1 (/M	100 MG 15 ML	100	MG	15	ML	GOTAS
434	PRODEJEMPLO_5 GRAG 10 MG x 15	10 MG	10	MG	-	-	GRAGEAS
460	PRODEJEMPLO_6 JBE 200 ML x 1	200 ML	200	ML	-	ML	JARABE

Ilustración 4: Caso Real - Primera parte del Layout

En la Ilustración 4 muestro parte del layout que era requerido para la fácil lectura de la información. Tomando en cuenta el comentario extra en la imagen anterior, la información de la columna *presentacion* necesitaba ser desplegada en las columnas

DESCRIP, CONCENTR_SOL, SOLIDO, CONCENTR_LIQ, LIQUIDO, PRESENTACIÓN de una forma más amplia para su mejor lectura.

		2014 Oct Units	2014 ... Units	2014 Dic Units	2014 Oct Values	2014 ... Values	2014 Dic Values	2015 Ene Units
descriptiva_7	descriptiva_8							
E	M							
T	N							
U	Y							
I	T							
T	N							
U	Y							

Ilustración 5: Caso Real - Segunda Parte del Layout

En la Ilustración 5 muestro la continuación del layout donde vienen especificadas el resto de las variables pedidas y la historia de la data requerida.

Una vez que tuve claros los requerimientos y las variables definidas comencé a elaborar las consultas en SQL para extraer la información solicitada.

- Pruebas realizadas

Dado que no tuve acceso a ningún diagrama o esquema para ver las relaciones entre las tablas, me di a la tarea de inferir dicho modelo con base en mi experiencia consultando las bases. Tuve que consultar todas las tablas involucradas en el reporte, acudí con personal del área para que pudiera orientarme sobre en qué tabla podía consultar lo que necesitaba. Tuve que ver los tipos de datos de las columnas e inferir qué columna podía servir como llave primaria.

Después de concluidas dichas pruebas, el modelo que inferí es el mostrado a continuación.

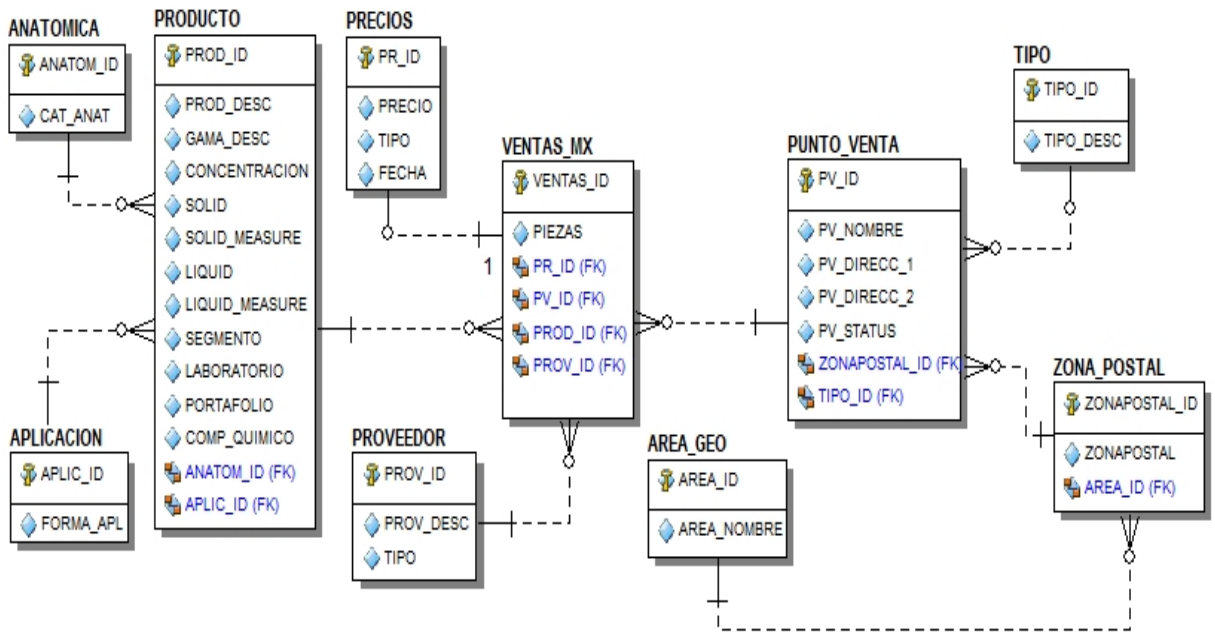


Ilustración 6: Modelo Dimensional del Data Warehouse

En la Ilustración 6 muestro el modelo dimensional del almacén de datos en el que realicé las consultas.

En este modelo presento la tabla de hechos llamada VENTAS_MX y el resto de las dimensiones. Por razones de privacidad, el nombre de las tablas y atributos fueron cambiados. También fueron omitidas ciertas tablas y atributos que no eran relevantes para el entendimiento del modelo en este informe.

Para elaborar el folio, dividí el código SQL en varias secciones que lo dimensionaban; en la primera trabajé todos los datos concernientes a las dimensiones espaciales, en la segunda aquellos referentes a los productos y en la tercera todo lo referente a la dimensión temporal y valores numéricos.

```
/*=====Reporte Ventas======*/

/*MAESTRO*/
select pv_id, tipo_desc,
case when tipo_id = 70 then 22
when tipo_id = 53 or tipo_id = 37 then 65
when tipo_id = 79 then 30
else tipo_id end 'Tipo_PDV'
into CLOPEZROA.MAESTRO_643
from DBO.PUNTO_VENTA a join DBO.TIPO b
on a.tipo_id = b.tipo_id
where tipo_desc = 'FARMACIAS' or tipo_desc = 'HOSPITALES'
```

Código 1: Maestro

En la Código 1 muestro la creación de la tabla maestro del folio correspondiente, en la cual almacené los datos espaciales requeridos. En este caso, solo necesité la información de todos los puntos de venta que hayan sido farmacias u hospitales.

```
/*MERCADO*/
select a.prod_id, b.prod_desc, a.comp_quimico, b.anatom_id
---Despliegue de la concentración dependiendo del caso
case when ((a.solid_measure = 'MG' or a.solid_measure = 'G')
     and a.liquid_measure = 'ML') then concat(a.solid, a.liquid)
     when ((a.solid_measure = 'MG' or a.solid_measure = 'G')
     and a.liquid_measure = 'L') then concat(a.solid, a.liquid)
     when ((a.solid_measure = 'MG' or a.solid_measure = 'G')
     and a.liquid_measure like '%?%') then a.solid
     when ((a.liquid_measure = 'ML' or a.liquid_measure = 'L')
     and a.solid_measure like '%?%') then a.liquid
     when a.solid_measure like '%?%' or a.liquid_measure like '%?%'
     then '-'
     end 'descrip',
case when ((a.solid_measure = 'MG' or a.solid_measure = 'G')
     and a.liquid_measure like '%?%') then a.solid
     when (a.liquid_measure = 'ML' or a.liquid_measure = 'L')
     then '-'
     end 'concentr_solido',
case when ((a.solid_measure = 'MG' or a.solid_measure = 'G')
     and a.liquid_measure like '%?%') then a.solid_measure
     when (a.liquid_measure = 'ML' or a.liquid_measure = 'L')
     then '-'
     end 'solido',
case when ((a.liquid_measure = 'ML' or a.liquid_measure = 'L')
     and a.solid_measure like '%?%')
     then a.liquid
     when (a.solid_measure = 'MG' or a.solid_measure = 'G')
     then '-'
     end 'concentr_liquido',
case when ((a.liquid_measure = 'ML' or a.liquid_measure = 'L')
     and a.solid_measure like '%?%') then a.liquid_measure
     when (a.solid_measure = 'MG' or a.solid_measure = 'G')
     then '-'
     end 'liquido',
--Termina separación
a.aplic_id as code_app, c.forma_aplc as desc_app,
a.segmento as sgmnt, a.laboratorio as lab
into CLOPEZROA.MERCADO_643
from DBO.PRODUCTO a join CLOPEZROA.MDO_ARMADO B
on a.prod_id = b.codigos join DBO.APLICACION C
on a.aplic_id = c.aplic_id
where a.anatom_id not like 'X%'
```

Código 2: Mercado

Con el Código 2 creé la tabla mercado, la cual contenía la información acerca de los productos. En ese folio, fue necesario darle un tratamiento previo a la información que contendría esta tabla, ya que tuve que generar nuevas columnas a partir de la información de la concentración de los productos, esto derivado de requerimientos adicionales que tenía la solicitud mostrados en la Ilustración 4.

```
/*Periodos*/
--201410
select pv_id, tipo_desc, Tipo_PDV, prod_id, prod_desc,
comp_quimico, anatom_id, code_app, desc_app, sgmnt, lab,
SUM(PIEZAS) PIEZAS, SUM(PIEZAS*PRECIO) VALORES, 201410 MES
into CLOPEZROA.FOLIO_643
from CLOPEZROA.MERCADO_643 A
join DBO.VENTAS_MX_201410 B ON a.prod_id = b.prod_id
join CLOPEZROA.MAESTRO_643 C on b.pv_id = c.pv_id
left join (select pr_id, convert(float,precio)/100 precio
from DBO.PRECIOS where tipo = 'T'
and date='20141001') D on b.pr_id = d.pr_id
join DBO.PROVEEDOR E on b.prov_id = e.prov_id
group by pv_id, tipo_desc, Tipo_PDV, prod_id, prod_desc,
comp_quimico, anatom_id, code_app, desc_app, sgmnt, lab

--201411
insert into CLOPEZROA.FOLIO_643
select pv_id, tipo_desc, Tipo_PDV, prod_id, prod_desc,
comp_quimico, anatom_id, code_app, desc_app, sgmnt, lab,
SUM(PIEZAS) PIEZAS, SUM(PIEZAS*PRECIO) VALORES, 201411 MES
from CLOPEZROA.MERCADO_643 A
join DBO.VENTAS_MX_201411 B ON a.prod_id = b.prod_id
join CLOPEZROA.MAESTRO_643 C on b.pv_id = c.pv_id
left join (select pr_id, convert(float,precio)/100 precio
from DBO.PRECIOS where tipo = 'T'
and date='20141101') D on b.pr_id = d.pr_id
join DBO.PROVEEDOR E on b.prov_id = e.prov_id
group by pv_id, tipo_desc, Tipo_PDV, prod_id, prod_desc,
comp_quimico, anatom_id, code_app, desc_app, sgmnt, lab
```

Código 3: Cruce Mensual

El Código 3 contiene bloques de código en donde hice el cruce de las tablas *MAESTRO* y *MERCADO* – creadas previamente – con las tablas de precios *DBO.PRECIOS* y proveedores *DBO.PROVEEDOR,* así como con la tabla que contiene las piezas vendidas *DBO.VENTAS_MX_<AAAMM>*.

Escribí un bloque por cada mes de acuerdo con el periodo definido en los requerimientos, en este caso, el periodo constaba de dos años hacia atrás a partir de octubre del 2016, por lo que repetí el mismo bloque de código 24 veces, pero variando el mes. De esta manera, extraje el número de piezas vendidas y los valores calculados a partir de los precios de forma mensual.

La tabla *DBO.VENTAS_MX_<AAAMM>* es la tabla de hechos que contenía las métricas, las cuales, en este caso, eran las piezas vendidas. Dicha tabla contenía las llaves primarias del resto de las tablas de dimensiones: MAESTRO (punto de venta), MERCADO (productos), Y PROVEEDOR.

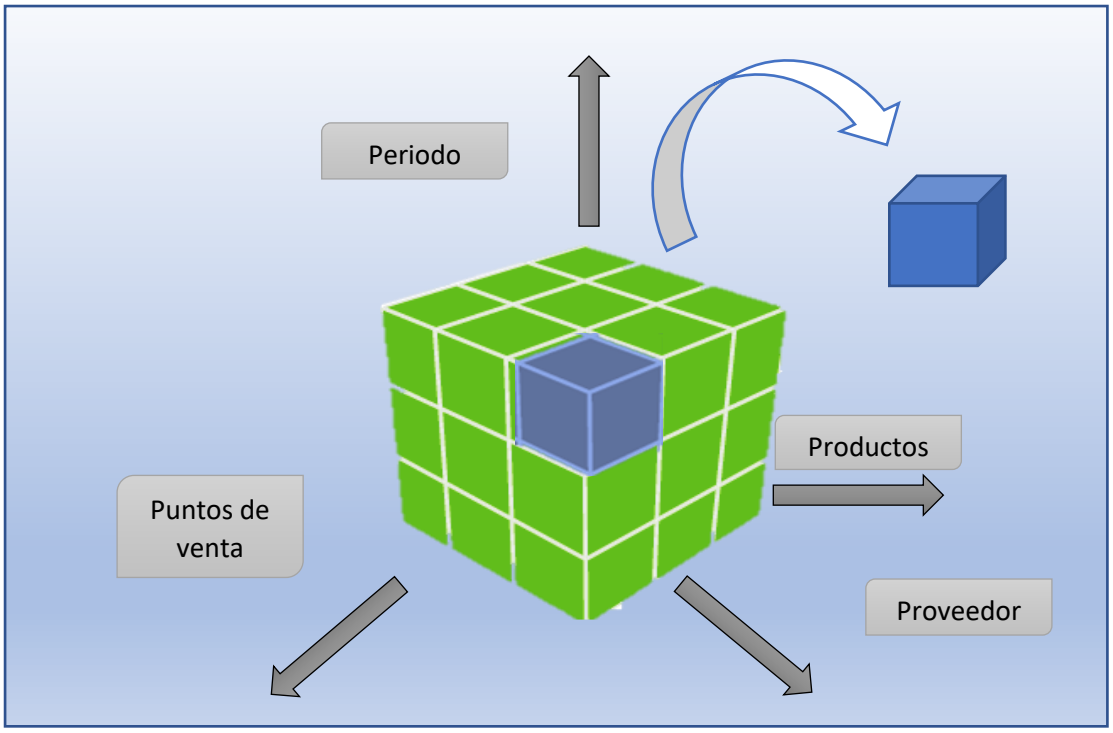


Figura 5: Cubo de Información

Para facilitar la comprensión de cómo los datos eran extraídos, en la Figura 5 muestro un cubo de información, el cual es una representación abstracta del modelo de datos consultado ya que muestra todas las dimensiones que componían dicho modelo.

En este caso, el modelo dimensional consta de 4 dimensiones que, al realizar "cortes" en los distintos ejes que representan a cada dimensión, podíamos obtener la información que deseábamos al mayor detalle posible.

Es decir, de esta forma se obtenían las piezas vendidas del producto W a través del proveedor X en el punto de venta Y durante el periodo Z. El cubo azul representa el resultado de una consulta de este tipo.

- Problemas presentados

Dado que todas las entregas debían realizarse en archivos Excel, tuve que prestar especial atención en aquellos folios que generaban un número de registros superior al millón, pues Excel no soportaba esa cantidad.

Inmediatamente después de que la consulta finalizaba, revisaba el conteo de registros totales y cuando éste superaba el millón me contactaba con el consultor a cargo de dicha petición para comentarle que su archivo tenía que ser segmentado y llegar a un acuerdo sobre cómo se iba a realizar dicha segmentación. En otras palabras, escogíamos las columnas "aptas" para seccionarlas sin modificar la fácil comprensión y análisis del archivo.

Después de esto, procedía a realizar una fragmentación horizontal de la tabla de acuerdo con los registros de la(s) columnas(s) seleccionada(s).

Dada una relación R, su fragmentación horizontal está definida como: [11]

$$Ri = \sigma_{Fi}(R)$$

Donde:
R es una relación
Ri es el predicado de fragmentación
Fi es la fórmula de selección usada para obtener Ri

Tabla_Final

PROD_DESC	LAB	FORMA_APL	MES	PIEZAS
PROD1	LAB1	SUBLINGUAL	201411	5
PROD2	LAB2	CUTANEA	201411	13
PROD3	LAB3	SUBLINGUAL	201411	2
PROD4	LAB1	OFTALMICA	201411	24
PROD5	LAB2	CUTANEA	201411	9
PROD6	LAB3	CUTANEA	201411	33
PROD7	LAB5	CUTANEA	201411	25
PROD8	LAB2	SUBLINGUAL	201411	7
……	……	…….	…….	……
PROD1	LAB1	SUBLINGUAL	201611	43
PROD2	LAB2	CUTANEA	201611	12
PROD3	LAB3	SUBLINGUAL	201611	5
PROD4	LAB1	OFTALMICA	201611	1
PROD5	LAB2	CUTANEA	201611	24
PROD6	LAB3	OFTALMICA	201611	37
PROD7	LAB5	CUTANEA	201611	5
PROD8	LAB2	SUBLINGUAL	201611	2

Fragmento_1 = σ FORMA_APL = 'Sublingual' (Tabla_Final)

Fragmento_2 = σ FORMA_APL = 'Cutánea (Tabla_Final)

Fragmento_2 = σ FORMA_APL = 'Oftálmica' (Tabla_Final)

Figura 6: Fragmentación de la Tabla

En la Figura 6 muestro la fragmentación horizontal de la tabla ejemplo TABLA_FINAL pues contenía un número de registros elevado el cual no era posible de entregar en un archivo Excel. La fragmentación la realicé mediante la columna FORMA_APL la cual contenía: sublingual, cutánea y oftálmica; de esta manera, entregué 3 archivos, cada uno con una forma de aplicación diferente. Por lo tanto, la lectura de los archivos fue posible sin haber alterado los datos.

4.1.1.3 Resultados obtenidos

El proceso de elaboración de folios varió en cuanto a las dimensiones y métricas requeridas de un folio a otro, pero las consultas las elaboré con flujos de trabajo muy similares a los expuestos en esta sección.

Elaboré alrededor de 90 folios pertenecientes a distintos proyectos por parte de consultoría. Con esto, participé en los estudios que esa área llevaba a cabo, en donde el cliente decidía si contrataba o no el servicio de la empresa.

4.1.2 Análisis de ventas

Dos de los tipos de data con las que cuenta la empresa son la venta de entrada y la venta de salida, dichas ventas tienen que presentar una tendencia muy similar para poder decir que los datos son procesados de forma correcta. Yo elaboré estos análisis, los cuales consistían en ejecutar consultas en SQL para posteriormente realizar cálculos estadísticos y gráficos que facilitaron la visualización de la información mediante scripts programados en lenguaje R y presentarlos a gerentes.

4.1.2.1 Antecedentes

Además del soporte dado a otras áreas, estadística se encargaba de realizar un estudio especial el cual era mostrado a los directivos cada trimestre. Dicho estudio consistía en una serie de análisis a nivel general de los dos tipos de ventas que se manejan dentro de la empresa: la venta de entrada y la venta de salida.

La venta de entrada – la cual llamaré VE – es la venta hecha de los proveedores hacia los puntos de venta. La venta de salida – llamada VS – es la venta de los puntos de venta al público.

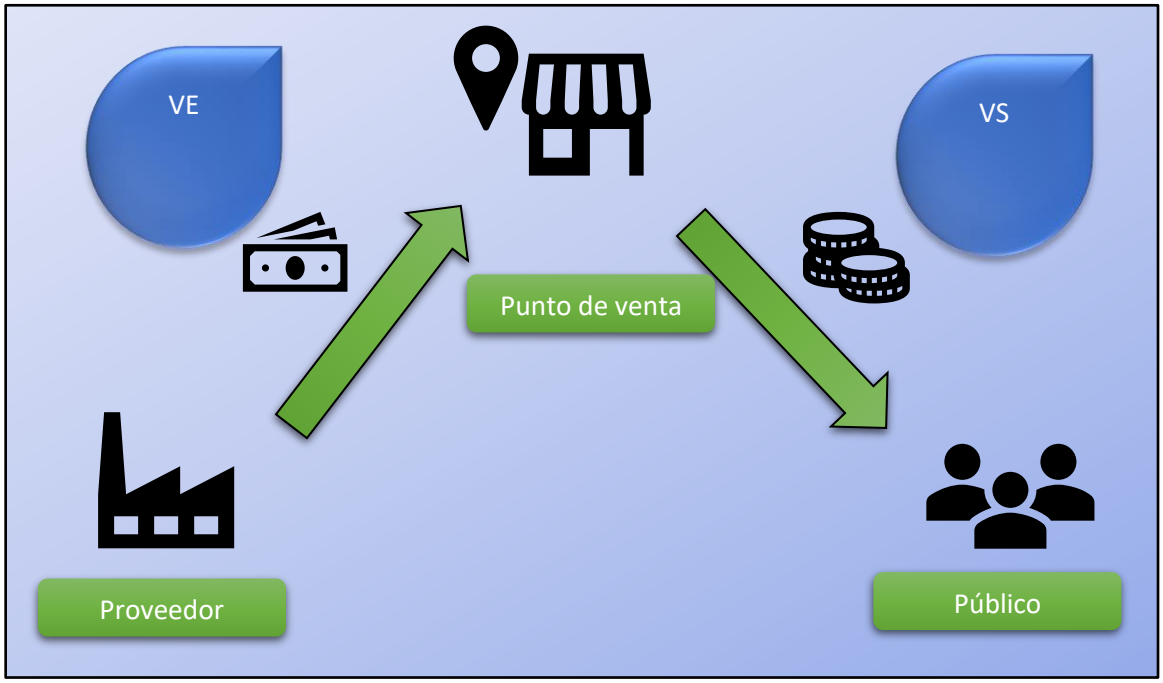


Figura 7: Tipos de Venta

En la Figura 7 muestro el flujo de algunos tipos de ventas que capta la empresa.

La finalidad de este estudio fue detectar anomalías referentes a la calidad de los datos que se procesan. Para poder indagar sobre esto fue necesario realizar el análisis a diferentes niveles de apertura de información, es decir, yendo de lo general a lo particular navegando por las dimensiones de los datos.

Por lo tanto, los análisis los realicé desde nivel laboratorio, bajando a nivel gama y llegando hasta nivel producto. Esto mismo lo llevé a cabo para los dos tipos de producto, el tipo A – aquellos cuya venta solo es posible con receta médica – y el tipo B – aquellos que se pueden adquirir en un punto de venta sin receta médica, tales como cremas, jeringas, jabones o preservativos –

4.1.2.2 Metodología de resolución

- Extracción de los datos

Para la extracción de la información elaboré scripts en SQL con la misma estructura a los ya descritos en los códigos 1,2 y 3 pues las bases de datos de donde consultaba esta información eran las mismas. Únicamente traje nuevas columnas útiles tales como LABORATORIO, GAMA_DESC y PROD_DESC para los análisis posteriores y separé los productos tipo A del tipo B. La métrica seguía siendo la misma, la cantidad de piezas vendidas.

- Preparación de los datos

Una vez que tenía los datos extraídos de las bases, los segmenté por tipo y los separé de acuerdo con la apertura, es decir, en un archivo Excel coloqué los datos donde la apertura máxima fuera el laboratorio, en otro en donde la apertura fuera la gama y en uno diferente donde fuera el producto. Por lo tanto, la cantidad de piezas totales no cambiaba ya que solo variaba la agrupación de los datos, no el cálculo de agregación.

Dado que el objetivo de estos análisis era comparar ambos tipos de venta, en cada archivo incluí una columna donde calculaba el r-value – formalmente llamado coeficiente de Pearson – el cual proporciona una medida numérica de la correlación entre dos variables y nos dice qué tan fuerte es esa relación. [12]
Por lo tanto, este valor, me decía que tan similar era la VE respecto a la VS.

Para interpretar el resultado correctamente bastaba con saber que, entre más próximo a 1 era dicho valor, la correlación era perfecta.

PROD_ID	PROD_DESC	GAMA_DESC	LAB	PZAS_VS	PZAS_VE	Rvalue
23754	PRODUCTO_23754	GAMA_23754	LAB A	33464580	31880206	1.049697734
188098	PRODUCTO_188098	GAMA_188098	LAB B	20038222	19690555	1.017656536
64468	PRODUCTO_64468	GAMA_64468	LAB C	19770839	20677225	0.956165008
25876	PRODUCTO_25876	GAMA_25876	LAB B	12987865	11286557	1.150737554
34679	PRODUCTO_34679	GAMA_34679	LAB D	9330917	9028105	1.033541037
509513	PRODUCTO_509513	GAMA_509513	LAB B	8436104	8767743	0.9621751
346011	PRODUCTO_346011	GAMA_346011	LAB Z	7727783	7608120	1.015728327
243567	PRODUCTO_243567	GAMA_243567	LAB B	7617780	7806455	0.975830899
354891	PRODUCTO_354891	GAMA_354891	LAB F	6130675	6297696	0.973479031
14535	PRODUCTO_14535	GAMA_14535	LAB B	5993577	6182499	0.969442454
198988	PRODUCTO_198988	GAMA_198988	LAB M	5563793	5429321	1.024767738

Ilustración 7: Preparación de los datos

La Ilustración 7 es un ejemplo del archivo donde calculaba el r – value a partir de las ventas VS y VE teniendo PROD_ID como máximo nivel de apertura.

A partir de estos archivos y de haber realizado distintos cruces de información creé gráficas en donde fue fácil ver el comportamiento de ambas ventas a lo largo del tiempo para detectar en qué meses dichas ventas rompían tendencia.

Por lógica del negocio la correlación de ambos tipos de venta debía ser muy fuerte (r – value ≈ 1.0) en un mismo mes para poder decir que los datos son correctos, es decir, ambos tipos de venta tenían que tener un número de piezas vendidas muy similar en el mismo periodo. Los archivos con esta información los llamé *archivos de tendencia*.

- Tratamiento de los datos

Ya que los datos se encontraban correctamente segmentados por apertura y concentrados con ambos tipos de venta, lo siguiente que hice fue manipularlos utilizando lenguaje R para obtener las anomalías desde los distintos niveles de apertura, esto con la finalidad de que la interpretación fuera más sencilla y rápida.

Desde R Studio importé los archivos Excel para poder trabajarlos y realizar los cálculos correspondientes que me permitieran detectar anomalías.

De forma general, en cada script de R, llevé a cabo lo siguiente

1- Importé el archivo con la apertura requerida.
2- Obtuve el porcentaje de participación de cada registro dado, es decir, calculé el cociente de las ventas de cada registro entre la suma de las ventas totales.
3- Obtuve la suma acumulada del porcentaje participación.
4- Definí rangos para clasificar a los registros por categorías. Las categorías estaban definidas de acuerdo con el peso de las piezas vendidas según el rango de apertura en cuestión.
5- Creé diagramas de caja – o boxplot – donde mostré las ventas distribuidas por categoría y el r – value correspondiente. En estos gráficos es posible apreciar las anomalías – datos fuera de una distribución – encontradas dentro de cada categoría.

```
##BOXPLOT - Product

# Se importa tabla de Excel
x = R_boxplot # tabla guardada en variable x
x$share = x$`U VentaEntrada`/sum(x$`U VentaEntrada`) #crear market share

x_ord = x[order(x$share,decreasing = TRUE),] ##Se ordena de mayor a menor

x_ord$cum_share = cumsum(x_ord$share)## se crea cumsum
x_ord$cat = NA #se crea columna y se rellena con NA

#Se llena la col cat dependiendo de qué condición se cumpla
x_ord$cat[x_ord$cum_share<=0.5] = 1
x_ord$cat[x_ord$cum_share>0.5 & x_ord$cum_share<=0.8] = 2
x_ord$cat[x_ord$cum_share>0.8 & x_ord$cum_share<=.95] = 3
x_ord$cat[x_ord$cum_share>0.95] = 4

# boxplot(x$`TOTAL  MESES`)
boxplot(x_ord$`TOTAL  MESES`~ x_ord$cat,ylim=c(0,2),
main="PRODUCT R-value by Cat",
xlab = "Categoria",ylab = "R-Value") ##boxplot de las 4 cat.

library(ggplot2)
fill <- "#0598FA"
line <- "#1F3552"
ggplot(x_ord, aes(x = as.factor(cat), y = `TOTAL  MESES`)) +
geom_boxplot(fill = fill, colour = line) +
scale_y_continuous(limits=c(0,2),  expand = c(0, 0))+
ggtitle("PRODUCT R-value by cat") + xlab("Cat") + ylab("R-Value")
```

Código 4: Generación de BoxPlot en Lenguaje R

En el código 4 muestro el que escribí en lenguaje R para poder clasificar los productos en 4 categorías de acuerdo con las ventas que tienen, de esta forma, en la categoría 1 se encontraban todos aquellos cuya suma de sus ventas conforma el 50% de la venta total, en la categoría 2 estaban aquellos cuya venta hace el 30%, en la categoría 3 los productos con una venta que cubre el 15% y en la categoría 4 aquellos cuya venta representa el 5% de la venta total.

En este código muestro también la generación los gráficos boxplot de las piezas vendidas contra el r – value de cada categoría.

Una vez que tuve conformadas las 4 categorías, lo siguiente fue detectar las anomalías en cada una de las aperturas, especialmente aquellos datos anómalos presentes a nivel producto. Las anomalías – o outliers – son valores individuales que quedan fuera del patrón general de los datos. [13]

Para lograrlo, y partiendo de los gráficos boxplot obtenidos anteriormente, calculé aquellos datos fuera del rango intercuartílico, es decir, aquellos valores que caían fuera de la distribución de los datos.

El rango intercuartílico IQR se define como una estimación estadística de la dispersión de una distribución de datos. Consiste en la diferencia entre el tercer y el primer cuartil. Mediante esta medida se eliminan los valores extremadamente alejados. [14]

IQR = Q3 – Q1
Donde:

Q3 = Tercer cuartil de la distribución.
Q1 = Primer cuartil de la distribución.

Ya que la definición de outliers es algo subjetiva utilicé una regla ya establecida que ayuda a determinar si un dato es realmente o no un valor anómalo.

Un dato es una potencial anomalía si y solo si [15]
- *Es menor a Q1 – (1.5*IQR)*
- *Es mayor a Q3 + (1.5*IQR)*

Donde:

IQR = Rango intercuartílico
Q3 = Tercer cuartil de la distribución.
Q1 = Primer cuartil de la distribución.

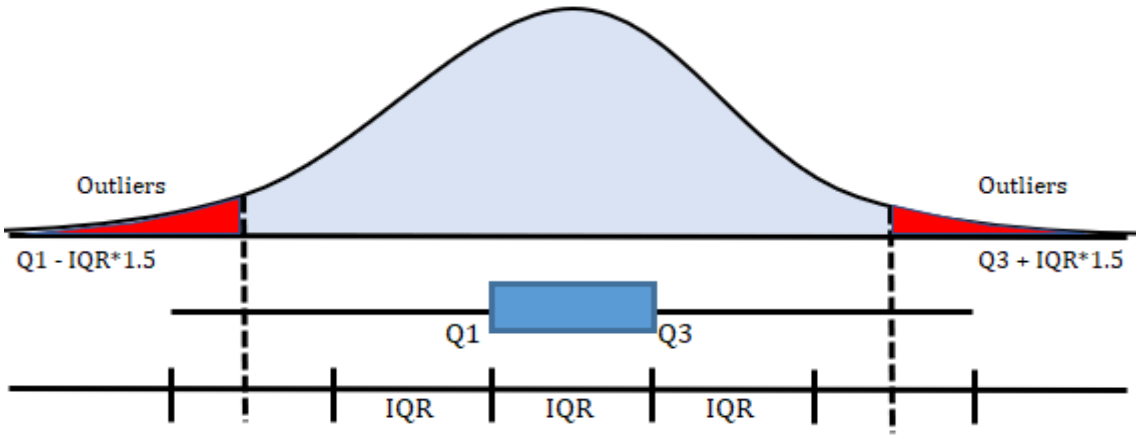


Figura 8: Rango Intercuartílico

La Figura 8 es la representación del boxplot, mostrando el rango intercuartílico, el primer y tercer cuartil, así como los valores extremos que serían considerados como outliers.

Apliqué esa misma regla a cada una de las categorías y fue posible detectar los outliers potenciales los cuales, a partir de ese momento, fueron considerados como alertas, pues no seguían el mismo patrón que el resto de los datos, y por lo tanto podían estar afectando la calidad de la información que se producía.

- Interpretación de la información

Cuando los archivos de tendencia y los boxplot estaban finalizados, el siguiente paso fue interpretar la información obtenida para poder cumplir con el objetivo inicial de estos análisis.

Las siguientes figuras que mostraré representan ejemplos reales de los gráficos e interpretaciones que llevé a cabo en las diferentes aperturas para poder obtener información que genere valor al negocio.

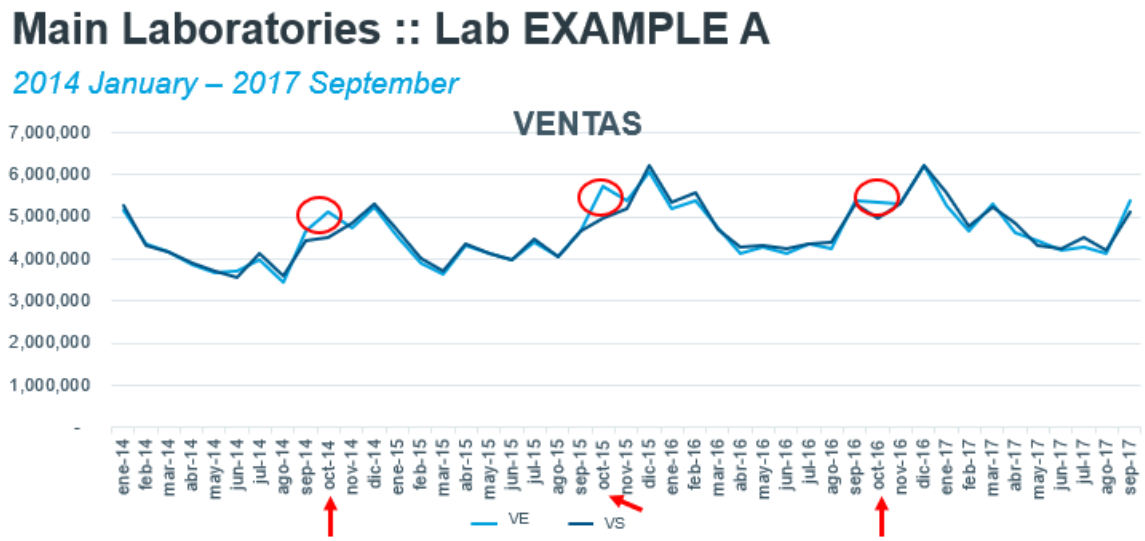


Ilustración 8: Ventas Laboratorio 1

En la gráfica de la Ilustración 8 se comparan los dos tipos de ventas totales de un solo laboratorio. Como se observa en los círculos rojos, la VE rompía la tendencia en repetidas ocasiones.

En un principio se pudo tratar de una alerta, pero los periodos en donde se presenta este "problema" eran los mismos cada año, por lo que pude concluir que esto no era debido a un problema en el procesamiento de los datos, sino que fue a causa de que este laboratorio se surtía por parte de los proveedores de manera masiva antes del periodo invernal, ya que los productos de este laboratorio son en su mayoría especiales para tratar enfermedades respiratorias provocadas por la temporada de invierno que estaba a punto de comenzar donde señalan las flechas rojas.

Para confirmar lo antes mencionado, se observa que en diciembre y enero de cada año sube la VS, fechas en donde la temporada invernal está en su mayor auge.

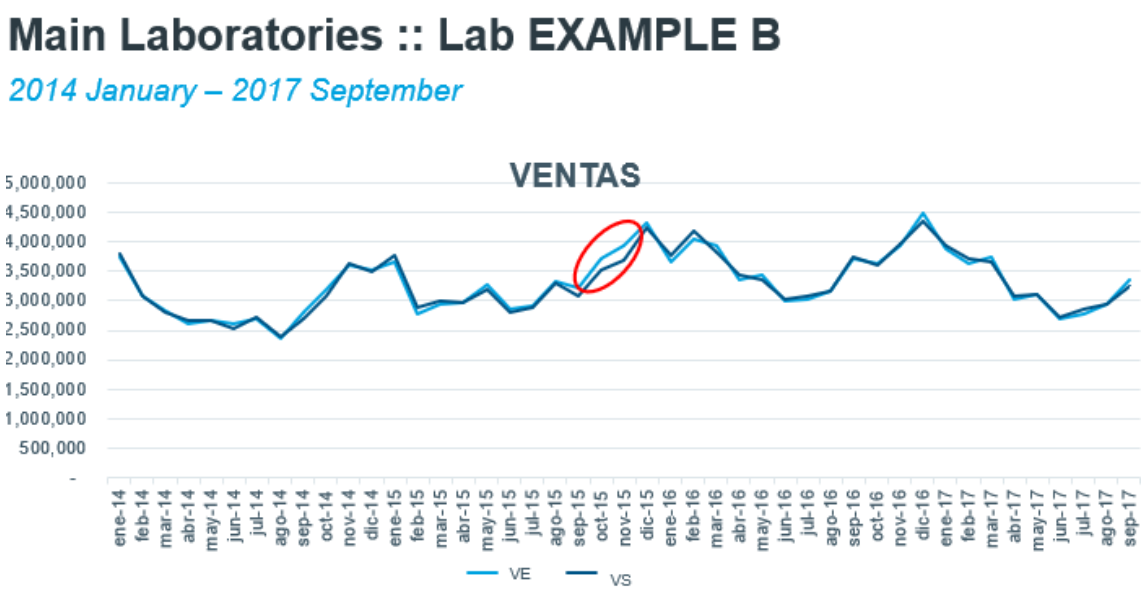


Ilustración 9: Ventas Laboratorio 2

Caso contrario al ejemplo anterior, en la Ilustración 9 muestro las ventas de otro laboratorio. Aquí podemos observar que la VE rompía tendencia durante un periodo de tres meses consecutivos y, además, este no era un patrón que se repetía. Por lo tanto, esto se convirtió en una alerta que debió ser tomada en cuenta para comenzar la corrección de la data histórica, ya que esto generaba datos incorrectos.

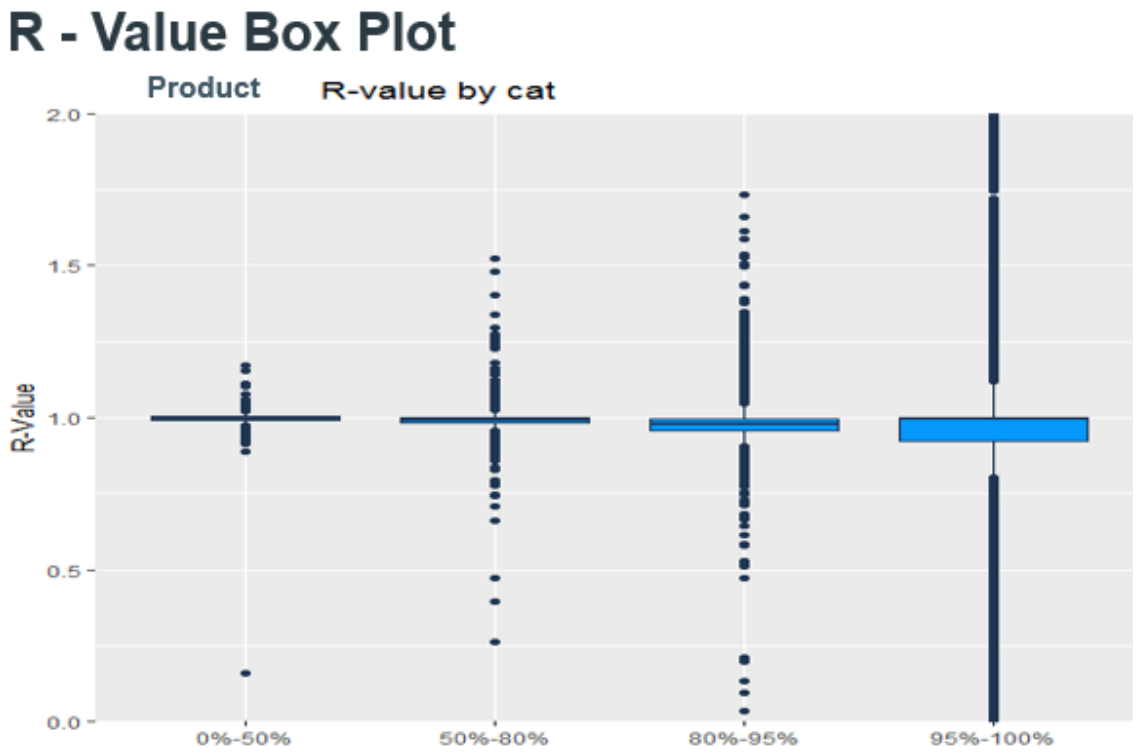
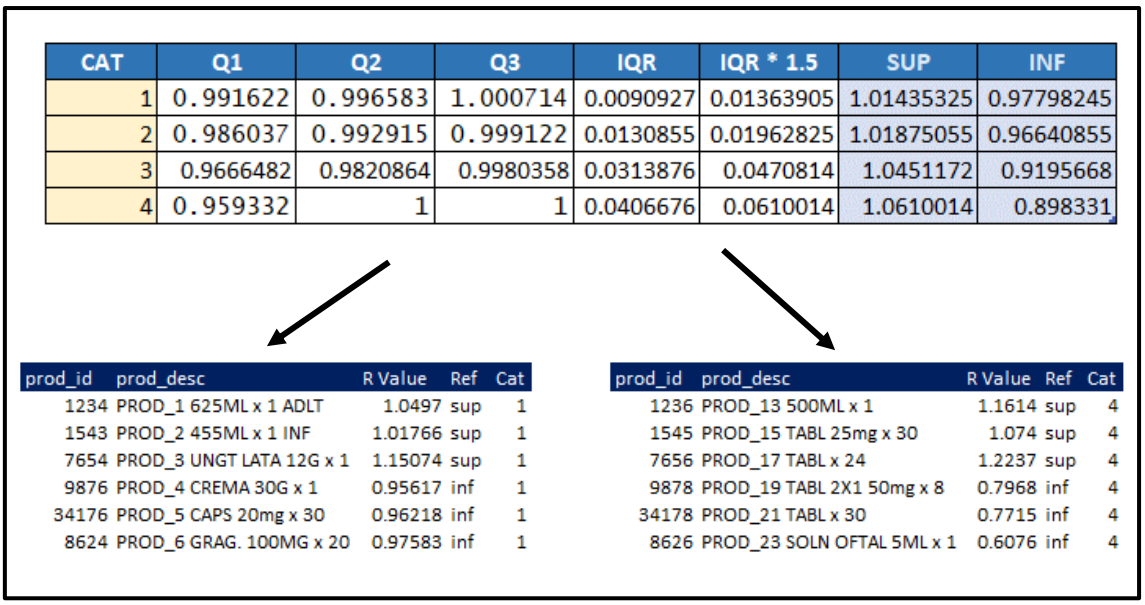


Ilustración 10: BoxPlot

Los diagramas boxplot de la Ilustración 10 son a nivel producto y los generé a partir del código explicado en el Código 4. Cada diagrama representa una categoría diferente.

El grosor de las cajas azules representa la mayor concentración de los datos, como podemos observar, los datos rondan el r – value = 1; los puntos alejados representan los outliers.

A partir del boxplot y realizando los cálculos de la regla para detectar outliers, generé la lista, por cada categoría, de los productos considerados como anomalías.



CAT	Q1	Q2	Q3	IQR	IQR * 1.5	SUP	INF
1	0.991622	0.996583	1.000714	0.0090927	0.01363905	1.01435325	0.97798245
2	0.986037	0.992915	0.999122	0.0130855	0.01962825	1.01875055	0.96640855
3	0.9666482	0.9820864	0.9980358	0.0313876	0.0470814	1.0451172	0.9195668
4	0.959332	1	1	0.0406676	0.0610014	1.0610014	0.898331

prod_id	prod_desc	R Value	Ref	Cat
1234	PROD_1 625ML x 1 ADLT	1.0497	sup	1
1543	PROD_2 455ML x 1 INF	1.01766	sup	1
7654	PROD_3 UNGT LATA 12G x 1	1.15074	sup	1
9876	PROD_4 CREMA 30G x 1	0.95617	inf	1
34176	PROD_5 CAPS 20mg x 30	0.96218	inf	1
8624	PROD_6 GRAG. 100MG x 20	0.97583	inf	1

prod_id	prod_desc	R Value	Ref	Cat
1236	PROD_13 500ML x 1	1.1614	sup	4
1545	PROD_15 TABL 25mg x 30	1.074	sup	4
7656	PROD_17 TABL x 24	1.2237	sup	4
9878	PROD_19 TABL 2X1 50mg x 8	0.7968	inf	4
34178	PROD_21 TABL x 30	0.7715	inf	4
8626	PROD_23 SOLN OFTAL 5ML x 1	0.6076	inf	4

Figura 9: Detección de Anomalías Categorizadas

En la Figura 9 muestro que, a partir del cálculo del IQR, establecí un umbral superior SUP y uno inferior INF por categoría con base al r – value de cada producto, con esto, todos los productos cuyo r – value fue superior al umbral SUP e inferior al umbral INF fueron considerados outliers.

Estos productos alertados fueron visibles como puntos fuera de la distribución en los diagramas de caja de la Ilustración 10.

4.1.2.3 Resultados obtenidos

Con los análisis que llevé a cabo en la sección anterior permití que mi área pudiera detectar problemas que impactan en la calidad de los datos que procesa la empresa.

Ya que todo el proceso lo comencé desde un nivel general – por laboratorio – y fui bajando hasta llegar a un mayor detalle – a nivel producto – se pudo recolectar una gran cantidad de información respecto a la calidad de los datos:

- Se detectó en qué laboratorios los datos no se están procesando correctamente.
- Fue posible determinar, de acuerdo con los patrones encontrados, si se trataba de un problema de datos erróneos o bien, si se trataba de un flujo de datos esperado del negocio.
- Se encontraron aquellos meses en donde las ventas VE y VS rompían tendencia.
- Se detectaron los productos con ventas cuyos valores eran atípicos, pues no entraban dentro de la distribución de valores del resto de los productos.

Gracias a la detección de todos estos puntos, fue posible que mi departamento alertara a los departamentos pertinentes dentro de la empresa para la pronta corrección y limpieza de estos datos, con el fin de que, para la siguiente producción de data, esta fuera correcta y no se siguieran arrastrando esas anomalías en las próximas producciones.

A grandes rasgos, el proceso que llevé a cabo fue el siguiente:

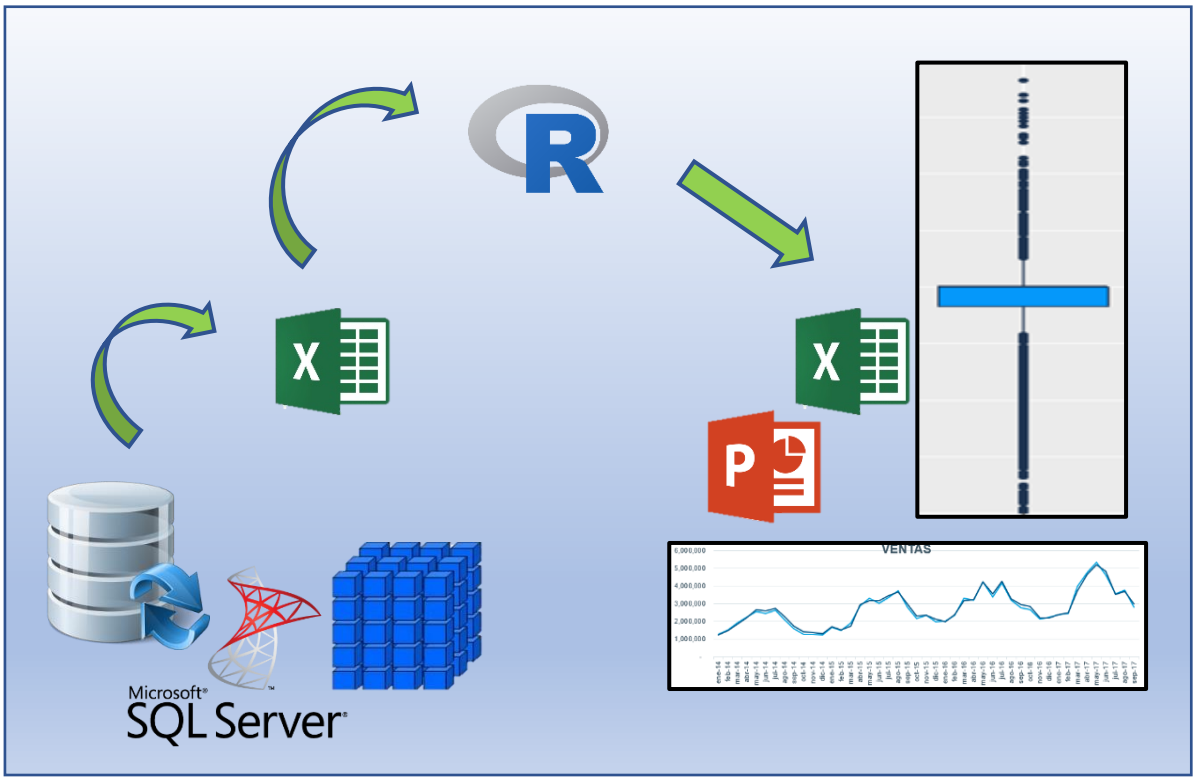


Figura 10: Proceso de Análisis de Ventas

En la Figura 10 muestro de manera general el proceso para elaborar este tipo de análisis, partiendo desde la extracción de los datos, pasando por el tratamiento de estos y finalizando con la interpretación de la información.

4.1.3 Optimización

Dado que el tiempo requerido para realizar los scripts en SQL de las consultas solicitadas era largo y, habiendo partes de código dentro del mismos scripts que no cambiaban y solo se repetían, programé procedimientos almacenados en donde solo era necesario introducir las variables solicitadas para que se ejecutara la consulta automáticamente.

4.1.3.1 Antecedentes

Partiendo de la sección 4.1.1.2 perteneciente a la metodología de elaboración de folios por peticiones del área de consultoría, fue necesario realizar ciertas modificaciones a los procesos internos de elaboración, ya que dichas peticiones comenzaron a ser más voluminosas debido a la necesidad de incluir una mayor cantidad de información histórica para análisis más robustos.

En consecuencia, las líneas de código de los scripts SQL también incrementaron, aumentando la probabilidad de errores humanos, por lo tanto, fue necesario automatizar dicho código con el fin de que su ejecución fuera más simple.

4.1.3.2 Metodología de resolución

Dado que las peticiones demandaban una mayor cantidad de información histórica, fue necesario modificar la obtención de los datos de esa dimensión con el fin de reducir las líneas de código ocupadas.

El código que optimicé fue el mostrado en el Código 3, el cual muestra dos bloques de código como ejemplo, uno consulta las ventas del mes de octubre y otro las de noviembre 2014, es decir, se generaba un bloque de código por cada mes a consultar.

El problema existía cuando los requerimientos demandaban alrededor de 60 meses de historia, lo cual significaba escribir ese mismo bloque de código 60 veces, pero variando las fechas para cada mes.

- Pruebas realizadas

Gracias al haber realizado una cantidad considerable de folios con anterioridad fue posible darme cuenta del tiempo de que me tomaba elaborar los scripts, así que comencé a recabar información referente a la cantidad de meses de historia que me solicitaban, así como también comencé a llevar un registro de las variables que normalmente nunca cambiaban, esto me permitió identificar qué parte del código era necesario automatizar.

Después de haber realizado las pruebas pertinentes y de haber conocido a detalle el proceso, me di a la tarea de programar un procedimiento almacenado para automatizar tal proceso. Un procedimiento almacenado es una agrupación de una o varias sentencias de SQL almacenada físicamente en la base de datos. [16]

El procediemiento almacenado que programé recibía como variables lo siguiente:

- Las dos tablas creadas anteriormente *maestro* y *mercado.*
- El nombre de tabla destino que se creará para almacenar el resultado.
- Una cadena de texto con las columnas a seleccionar de acuerdo a la información solicitada.
- El mes inicio y fin que conforman el periodo que se desea consultar.

1 . El procedimiento comienza por crear la tabla desitino y ejecuta el primer cruce de tablas, es decir, realiza joins entre las tablas maestro, mercado, la tabla de proveedores, la de precios y la tabla de ventas del mes ingresado como ***mes inicio.***

2. A continuación la fecha de inicio incrementa una unidad, lo cual genera el nombre de la tabla de ventas del mes consecutivo y realiza lo mismo que el paso anterior, es decir, hace un join entre todas las tablas con la tabla de ventas del ***mes inicio* + 1.**

3. El paso anterior se repite ya que la fecha de inicio continua incrementando una unidad y en cada incremento realiza el join corresponidentes. El incremento se detendrá cuando el ***mes inicio + 1*** sea igual al ***mes fin***.

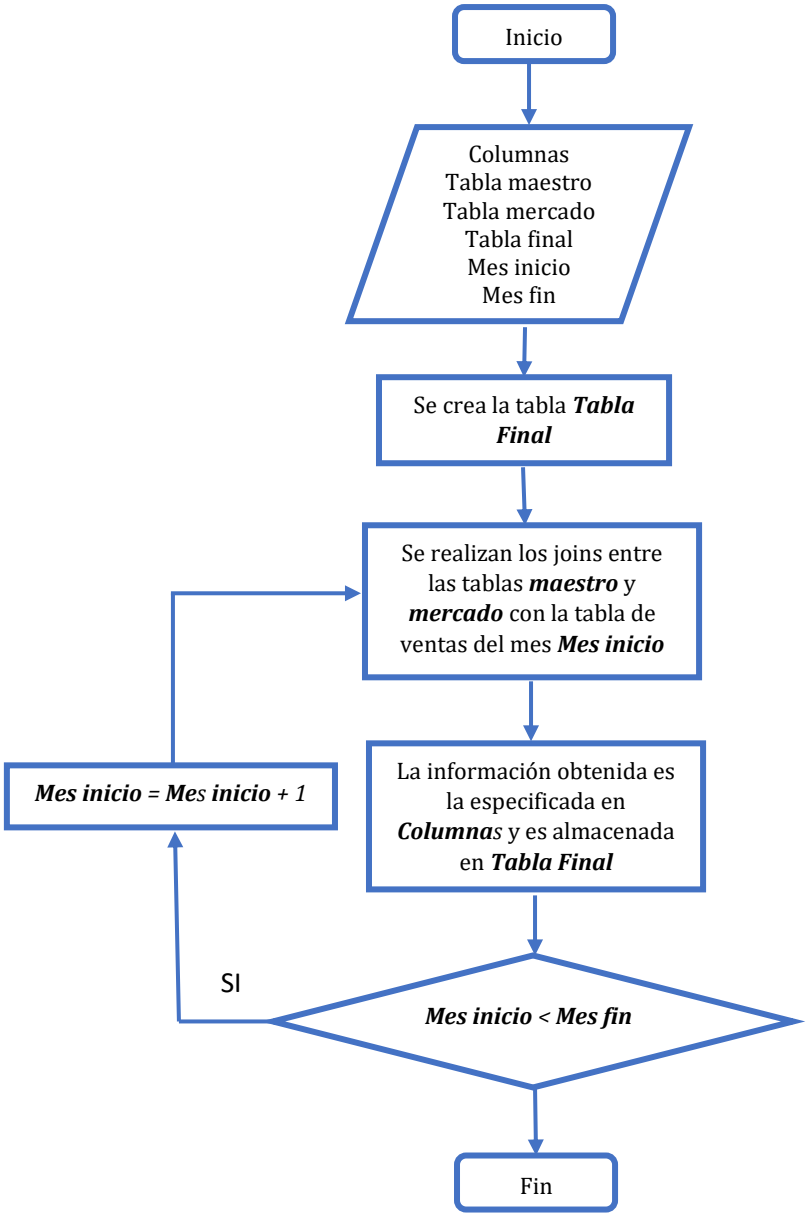


Figura 11: Diagrama de Flujo de SP_FOLIOS

La figura 11 muestra el diagrama de flujo del procedimiento almacenado que consulta todos los meses de un periodo especificado.

```
ALTER proc [clopezroa].[sp_folios]
@columnas varchar (300),
@nfolio varchar(60), --nombre de la tabla nueva que se generará
@nmaestro varchar(60), -- nombre de la tabla maestro
@nmercado varchar(60), -- nombre de la tabla mercado
@date1 int, -- fecha de inicio. AAMM
@date2 int -- fecha de fin. AAMM
as
          declare @x varchar(10)
          set @x = convert(varchar, @date1)
          exec (
                    ' select '+@columnas+', sum(piezas) PIEZAS,'+
                    ' sum(piezas*(convert(float,precio)/1000)) VALORES, 20'+@date1+
                    ' MES into '+@nfolio+' from '+@nmaestro+' a'+
                    ' join DBO.VENTAS_MEX_20'+@date1+' b on a.pv_id = b.pv_id'+
                    ' join '+@nmercado+' c on c.prod_id = b.prod_id'+
                    ' join DBO.PRECIOS d on d.pr_id = c.pr_id and fecha = '+''''+'20'+@x+'01'+''''+
                    ' join DBO.PROVEEDOR AS E on e.prov_id = b.prov_id
                   group by '+@columnas
               )

          declare @v int, @w int, @y int, @z varchar(10)
          set @v = @date1
          set @w = @date2
          set @y=@y+1
          set @z = convert(varchar,@y)
          exec(
                  'insert into '+@nfolio+
                  ' select '+@columnas+', sum(piezas) PIEZAS,'+
                  ' sum(piezas*(convert(float,precio)/1000)) VALORES, 20'+@v+
                  ' MES from '+@nmaestro+' a'+
                  ' join DBO.VENTAS_MEX_20'+@v+
                  ' b on a.pv_id = b.pv_id'+
                  ' join '+@nmercado+' c on c.prod_id = b.prod_id'+
                  ' join DBO.PRECIOS d on d.pr_id = c.pr_id'+
                  ' and fecha = '+''''+'20'+@z+'01'+''''+
                  ' join DBO.PROVEEDOR AS E on e.prov_id = b.prov_id
                   group by '+@columnas
              )
     end
     else
     begin
          set @v=@v+88
          set @y=@y+88
          set @z = convert(varchar,@y)
     end
   end
```

Código 5: SP_FOLIOS

El Código 5 muestra procedimiento almacenado SP_FOLIOS explicado en este punto.

4.1.3.3 Resultados obtenidos

Gracias a la automatización de esta tarea fue posible acelerar la entrega de los folios generados, pues con esto se disminuyó el trabajo manual que consistía en repetir un mismo bloque de código cierta cantidad de veces variando las fechas en diferentes tablas.

Además de lo mencionado, se disminuyó la probabilidad de obtener errores humanos provenientes de los cambios manuales de fecha, por ejemplo, puede darse el caso que

en la tabla de piezas vendidas se consulten las de 201403 (marzo 2014) pero que en la tabla de precios se teclee – por error – 201304 (abril 2013).

Con esto estaríamos consultando los precios de un mes diferente al de las piezas vendidas lo cual ocasionaría un error en la consulta y que, además, sería imposible detectarlo de manera inmediata pues, sintácticamente, no habría error alguno dado que el manejador no lo detectaría.

Hasta esta sección he descrito mis funciones como analista de datos en el área de estadística, a partir de aquí comenzaré a explicar mis actividades dentro del departamento de servicio al cliente.

4.2 Como analista de Servicio al Cliente

4.2.1 Explotación de Bases de Datos

Dado el acceso que tengo para poder consultar los datos directamente desde las bases - sin utilizar ninguna de las plataformas usadas - mi función principal fue atender solicitudes del resto de mi equipo respecto a anomalías o dudas que ellos mismos o el cliente detectaban, con el fin de justificar, alertar, o bien, corregir dichas anomalías. Las consultas que elaboré son muy similares a las elaboradas en mi anterior puesto con la diferencia que los requerimientos en estos casos eran a mayor detalle.

4.2.1.1 Antecedentes

El departamento de servicio al cliente, como bien lo dice su nombre, es el rostro de la empresa que hace frente a cada una de las peticiones o proyectos solicitados por parte de los diferentes laboratorios que conforman la plantilla de clientes.

Un cliente es un laboratorio que tiene contratado alguno de los servicios que la empresa ofrece, como lo son las auditorias de información. Una auditoria de información es una manera de monitorear y estructurar todos los datos de ciertos productos de interés para el laboratorio bajo distintas dimensiones, con el fin de obtener información que genere valor al negocio de cada laboratorio.

Dentro de las múltiples actividades que desarrollé en el área de servicio al cliente, una de las principales consistía en atender peticiones del resto de mi equipo para
la resolución de dudas acerca de anomalías o inconsistencias que impactaban negativamente la comprensión de la información. Dichas dudas podían provenir de los miembros de mi equipo – los ejecutivos de cuenta – o bien, directamente del cliente mismo.

Dado que el perfil profesional del personal de esta área estaba más enfocado al conocimiento de la industria farmacéutica, y a una alta capacidad de negociación, comunicación y análisis frente al cliente, yo como analista de datos contaba con las habilidades tecnológicas para explotar las bases de datos y obtener información útil que el cliente solicitaba.

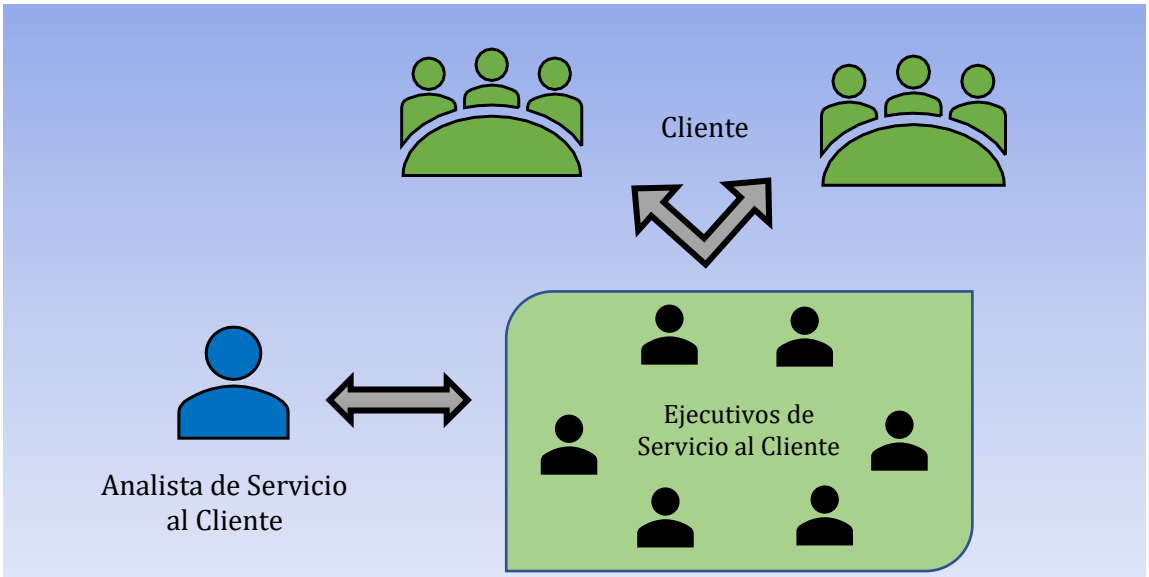


Figura 12: Estructura del Área de Servicio al Cliente

En la Figura 12 muestro la estructura interna del área. Mi función fue dar soporte concerniente al análisis de los datos al resto del equipo, ya que ellos eran los encargados de presentar tales análisis frente a los clientes.

4.2.1.2 Metodología de resolución

Los ejecutivos de cuenta y los clientes tenían acceso a la información de todas las auditorias que tenían contratadas a través de las distintas plataformas de BI, en las cuales podían visualizar distintos gráficos y reportes para monitorear dicha información.

Cuando ciertos datos no tenían sentido, generaban dudas respecto a su veracidad o definitivamente eran erróneos, me solicitaban generar reportes de información extraídos directamente del DWH para aclarar esas interrogantes, ya que dichas plataformas se alimentaban directamente del DWH mediante procesos ETL.

Los errores o anomalías encontrados eran consecuencia del procesamiento de los datos al realizarse la carga hacia las plataformas de BI. Aunado a esto, al ser capaz de explotar información directamente DWH, era posible tener acceso a todas las dimensiones de los datos, permitiéndome generar información desde una perspectiva más detallada, pues no todas las dimensiones de la información estaban disponibles a través de dichas plataformas.

Los reportes solicitados eran pedidos bajo detallados parámetros y dimensiones para que la respuesta que le tuvieran que dar al cliente fuera clara y concisa.

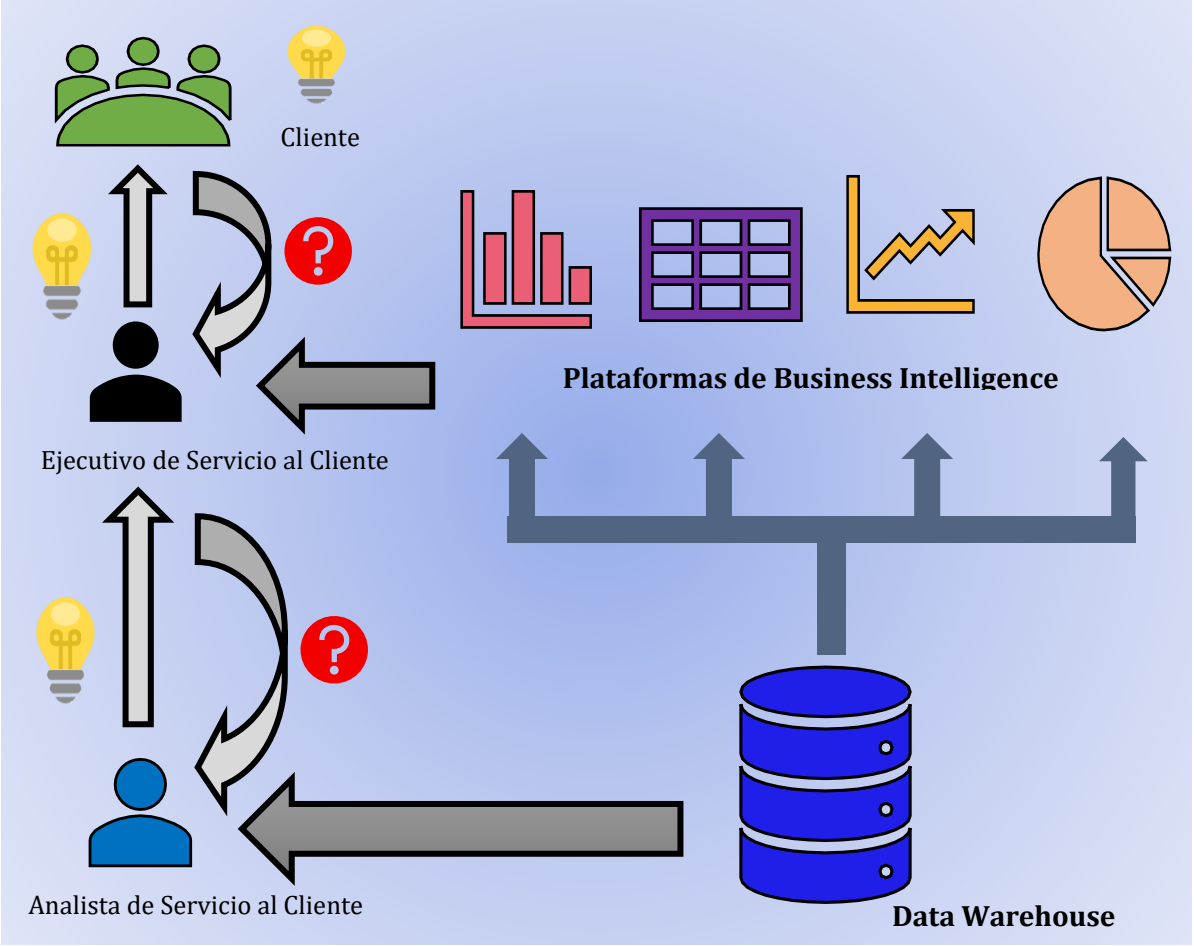


Figura 13: Flujo de Datos y Trabajo

En la Figura 13 muestro cómo las distintas plataformas de BI que utiliza el equipo de servicio al cliente se alimentan del DWH para formar las auditorias de datos que los clientes tienen contratadas.

Además, muestro cómo se obtenía la información: los ejecutivos y los clientes la visualizaban a través de las plataformas, yo, por mi parte, siendo analista, la obtenía realizando queries al DWH; de esta forma, cuando existían interrogantes acerca de la información que se estaba analizando, me solicitaban validar dicha información directamente de la fuente, ya sea en forma de peticiones o, directamente en reuniones con el cliente.

- Consulta caso real

A continuación presento un ejemplo de un tipo de consulta que realicé para aclarar cuestiones por parte del cliente en una reunión.

El cliente QWERTY tenía un portafolio de mercados, cada mercado estaba conformado por una lista de productos clasificados según el interés del cliente. QWERTY podía visualizar las ventas de sus mercados a nivel zona postal porque es así como lo tenía contratado, de acuerdo con las ventas de cada zona postal, él realiza ajustes financieros internos a su negocio.

Después de haberle entregado su data como usualmente se hacía, él se percató que repentinamente tuvo un incremento de ventas del 90% en una misma zona postal

comparado con el mes anterior. Esto generó dudas y me realizaron una petición para poder darles una respuesta.

Mi función fue traducir dicha petición de un lenguaje de negocio a un lenguaje técnico para poder hacer un query al DWH.

El código de las consultas era muy similar a las que realizaba en mi puesto anterior en el área de estadística, pues eran las mismas bases de datos.

```
/*MAESTRO*/
select pv_id, pv_nombre, pv_direcc_1, pv_direcc_2, zonapostal_id
into #maestro
from DBO.PUNTO_VENTA a join DBO.ZONA_POSTAL b
on a.zonapostal_id = b.zonapostal_id
where b.zonapostal_id = 'ZP9423'
/*MERCADO*/
select prod_id, prod_desc, laboratorio
into #mercado
from DBO.PRODUCTO
where prod_desc in ('prod1','prod2','prod3','prod4','prod5')
--201807
select pv_id, pv_nombre, pv_direcc_1, pv_direcc_2, zonapostal_id,
prod_id, prod_desc, laboratorio,
SUM(PIEZAS) PIEZAS, SUM(PIEZAS*PRECIO) VALORES, 201807 MES
into #QWERTY
from #mercado A
join DBO.VENTAS_MX_201410 B ON a.prod_id = b.prod_id
join #maestro C on b.pv_id = c.pv_id
left join (select pr_id, convert(float,precio)/1000 precio
from DBO.PRECIOS where tipo = 'T'
and fecha ='072018') D on b.pr_id = d.pr_id
join DBO.PROVEEDOR E on b.prov_id = e.prov_id
group by pv_id, pv_nombre, pv_direcc_1, pv_direcc_2, zonapostal_id,
prod_id, prod_desc, laboratorio
```

Código 6: Script Consulta a Cliente

En el Código 6 presento el script SQL que utilicé para realizar algunas de las consultas que me solicitaban de este tipo. En este ejemplifico el caso del cliente QWERTY el cual necesitaba saber la razón del incremento de sus mercados los cuales incluían a los productos: *prod1, prod2, prod3, prod4* y *prod5*, en la zona postal *ZP9423* en el mes de *julio 2018*.

Al obtener la consulta de datos al máximo nivel – por puntos de venta – me percaté que la razón fue debido a la apertura de nuevas sucursales de farmacias dentro de esa zona, no a un error en el procesamiento de los datos.

4.2.1.3 Resultados obtenidos

Con la pronta respuesta a los clientes de mi parte acerca de dudas referentes a la información que ellos recibían, contribuí a alcanzar y mantener los estándares de calidad de servicio existentes en la empresa, ya que la atención a cualquier petición del cliente no debía sobrepasar las 24 horas.

De igual forma, al atender estas peticiones participé en la mejora de los procesos internos, pues estos fluyeron con mayor rapidez, ya que el resto del equipo podía alertar a las áreas pertinentes acerca de potenciales errores o alertas en los datos antes

de que estos fueron cargados a las plataformas BI, evitando así, que errores de datos llegaran a manos del cliente.

4.2.2 Automatización para corrección de inconsistencias

Para poder obtener directamente la venta de salida de las bases de datos era necesario seguir determinados procesos y reglas de negocio para extraer la información correcta y evitar inconsistencias en el cálculo de ciertos datos. Mi función fue programar dichas reglas de negocio en procedimientos almacenados para su inmediata ejecución con el fin de agilizar el tiempo de los demás procesos a la hora de consultar este tipo de información.

4.2.2.1 Antecedentes

Diferentes áreas dentro de la empresa colaboraban en conjunto para realizar la producción y posterior generación de los datos que se liberan mensualmente, específicamente de los datos referentes a la VE y VS, pero, derivado de algunas reglas de negocio fue necesario ejecutar algunos procesos después de la producción mensual de la data para evitar inconsistencias en la misma.

Por su parte, los departamentos de Producción de Datos y Tecnología y Aplicaciones realizaban la corrección de estas inconsistencias para que los datos que visualizaban los clientes fueran correctos, pero de forma interna, en las bases de datos que utilizaba el departamento de Servicio al Cliente, fue necesario codificar procesos para realizar esta corrección y, que los datos de todas las fuentes de información que manejaba la empresa, cuadraran entre sí.

Una de mis responsabilidades fue realizar la programación – directamente en las bases de datos – de estas reglas de negocio para las correcta construcción y generación de la VS, evitando la propagación de tales inconsistencias en los reportes y análisis que solicitaban los clientes a mi departamento.

Las reglas de negocio a automatizar, de formar general, consistían en lo siguiente:

El objetivo era tener ambas ventas, VE y VS, de cada uno de los proveedores a lo largo de los 60 meses de historia con los que se cuenta, para que de esta forma se pudiera considerar la producción de dicha data como exitosa, pero derivado de procesos previos, la VS en algunos meses era igual a cero, por lo que se debía buscar en catálogos maestros los códigos equivalentes de dichos proveedores para poder insertar la venta correcta en el mes donde sea nula.

4.2.2.2 Metodología de resolución

- Pruebas realizadas

Para realizar esta automatización primero tuve que generar varias consultas a las bases para poder tener un perfilado de datos y así entender el problema de raíz, además, fue de mucha ayuda la experiencia consultando dichas tablas ya que son las mismas que he descrito en este informe, por lo tanto, el modelo dimensional es el mismo mostrado en la Ilustración 6.

Realicé múltiples consultas a las bases con el fin de replicar los errores y detectar en donde tenían que realizarse las correcciones, acudí con personal de distintas áreas para

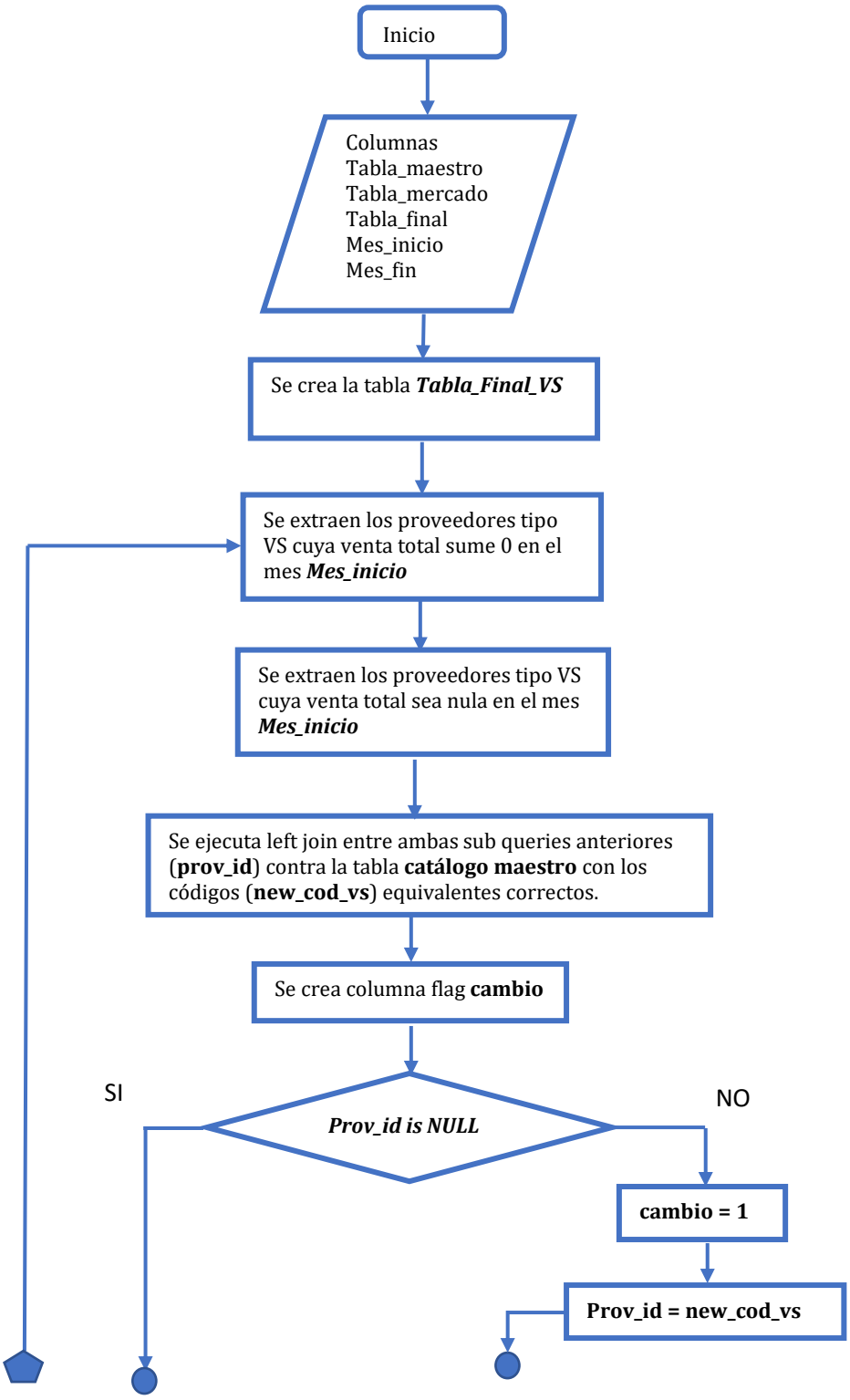
que me orientara en la comprensión de la información y en el origen de cada uno de los datos relevantes.

Después de haber realizado las pruebas necesarias y, dado que las reglas tenían que ser aplicadas directamente en las bases de datos, programé en un procedimiento almacenado que automatizara dichas reglas.

El procediemiento almacenado recibe como variables lo siguiente:

- Dos tablas creadas con información de puntos de venta y productos: *maestro* y *mercado.*
- El nombre de tabla destino que se creará para almacenar el resultado.
- Una cadena de texto con las columnas a seleccionar de acuerdo a la información solicitada.
- El mes inicio y fin que conforman el periodo que se desea consultar.

Programé este procedimiento lo más similar posible al anterior explicado en el código 5 con la finalidad de que no genere confusión para los usuarios que los utilicen.


Inicio
Columnas
Tabla_maestro
Tabla_mercado
Tabla_final
Mes_inicio
Mes_fin
Se crea la tabla *Tabla_Final_VS*
Se extraen los proveedores tipo VS cuya venta total sume 0 en el mes *Mes_inicio*
Se extraen los proveedores tipo VS cuya venta total sea nula en el mes *Mes_inicio*
Se ejecuta left join entre ambas sub queries anteriores (**prov_id**) contra la tabla **catálogo maestro** con los códigos (**new_cod_vs**) equivalentes correctos.
Se crea columna flag **cambio**
SI
Prov_id is NULL
NO
cambio = 1
Prov_id = new_cod_vs

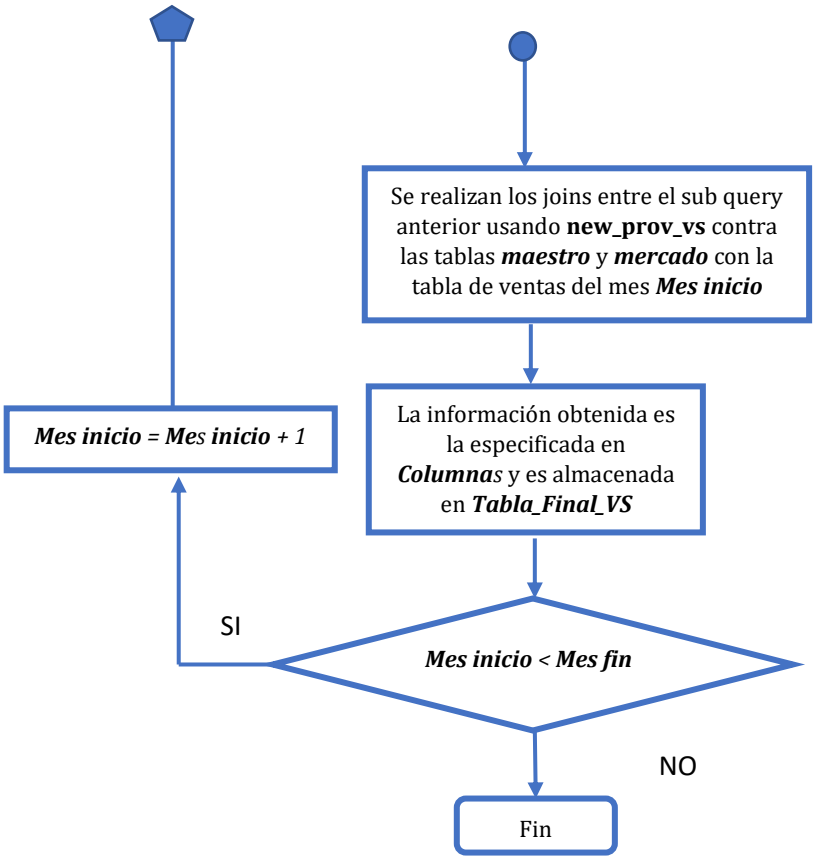


Figura 14: Diagrama de Flujo SP_VENTA_VS

La Figura 14 contiene el diagrama de flujo programado en el procedimiento almacenado *sp_venta_vs* que realizaba la búsqueda en el catálogo maestro de las ventas correctas por proveedor de todos aquellos que tienen venta mensual igual cero, para luego insertar esa venta en el mes correspondiente y continuar con los demás cruces en las tablas maestro, mercado y ventas del mes para generar la consulta requerida en un periodo específico.

```
ALTER proc [dbo].[sp_venta_vs]
@columnas varchar (300),
@nfolio varchar(40), --nombre de la tabla nueva que se generará
@nmaestro varchar(40), -- nombre de la tabla maestro
@nmercado varchar(40), -- nombre de la tabla mercado
@date1 int, -- fecha de inicio. AAMM
@date2 int -- fecha de fin. AAMM
as

declare @x varchar(10)
set @x = convert(varchar, @date1)
exec (
' select '+@columnas+', sum(units) UNIDADES, sum(units*(convert(float,price)/100)) VALORES, 20'+@date1+
' PERIODO into '+@nfolio+' from '+@nmaestro+' a'+
' join DBO.VENTAS_MX_20'+@date1+' b on a.PV_ID = b.PV_ID'+
' join '+@nmercado+' c on c.PROD_ID = b.PROD_ID'+
' left join DBO.PRECIOS d on d.PR_ID = b.PR_ID and date = '+''''+'20'+@x+'01'+''''+
' join  (
      select *,
      case when cambio = '+''''+'1'+'''' +'then VE else PROV_ID end new_cod_VS,
      case when cambio = '+''''+'1'+'''' +'then VE_desc else PROV_DESC end new_desc_VS
      from DBO.PROVEEDOR M
      left join (
           select VE, VE_desc, VS, ''1'' cambio
           from (
                 select VE.PROV_ID as VE, VE.PROV_DESC as VE_desc, VE.flag_X as VS, VS.PROV_DESC as VS_desc
                 from DBO.PROVEEDOR_MASTER as VE join DBO.PROVEEDOR_MASTER as VS
                 on VE.flag_X = VS.PROV_ID
                 where VE.tipo in (1)
                 and VE.flag_X is not NULL
                ) J join (
                          select distinct cod_VS
                          from DBO.VENTAS_MX_20'+@date1+' A
                          right join (
                                      select PROV_ID as cod_VS
                                      from DBO.PROVEEDOR_MASTER
                                      where tipo = VS
                                      ) B on A.PROV_ID = B.cod_VS
                           where PROV_ID is NULL

                           union all

                           select E.*
                           from
                             (
                              select NULL C1, PROV_ID
                              from (
                                    select PROV_ID, sum(units) suma0
                                    from DBO.VENTAS_MX_20'+@date1+
                                    ' group by PROV_ID
                                    ) C
                              where suma0 = 0
                              ) D join (
                                        select PROV_ID as cod_VS
                                        from DBO.PROVEEDOR_MASTER
                                        where tipo = VS
                                       ) E on D.PROV_ID = E.cod_VS
                ) K on J.VS = K.cod_VS --nulos y vneta 0
            ) N on M.PROV_ID = N.VS
       ) p
on PROV_ID = p.new_cod_VS
group by '+@columnas
)

declare @v int, @w int, @y int, @z varchar(10)
set @v = @date1
set @w = @date2
set @y = @date1
 while @v < @w
    begin
       if(right(@y,2)<='11')
       begin
           set @v=@v+1
           set @y=@y+1
           set @z = convert(varchar,@y)
```

```
exec(
'insert into '+@nfolio+
' select '+@columnas+', sum(units) UNIDADES, sum(units*(convert(float,price)/100)) VALORES, 20'+@v+
' PERIODO from '+@nmaestro+' a'+
' join DBO.VENTAS_MX_20'+@v+' b on a.PV_ID = b.PV_ID'+
' join '+@nmercado+' c on c.PROD_ID = b.PROD_ID'+
' left join DBO.PRECIOS d on d.PR_ID = b.PR_ID and date = '+''''+'20'+@z+'01'+''''+
' join (
     select *,
     case when cambio = '+''''+'1'+'''' +'then VE else PROV_ID end new_cod_VS,
     case when cambio = '+''''+'1'+'''' +'then VE_desc else PROV_DESC end new_desc_VS
     from DBO.PROVEEDOR M
     left join (
              select VE, VE_desc, VS, ''1'' cambio
              from (
                     select VE.PROV_ID as VE, VE.PROV_DESC as VE_desc, VE.flag_X as VS, VS.PROV_DESC as VS_desc
                     from DBO.PROVEEDOR_MASTER as VE join DBO.PROVEEDOR_MASTER as VS
                     on VE.flag_X = VS.PROV_ID
                     where VE.tipo in (1)
                     and VE.flag_X is not NULL
                    )  J join (
                              select distinct cod_VS
                              from DBO.VENTAS_MX_20'+@v+' A
                              right join (
                                        select PROV_ID as cod_VS
                                        from DBO.PROVEEDOR_MASTER
                                        where tipo = VS
                                       ) B on A.PROV_ID = B.cod_VS
                               where PROV_ID is NULL

                               union all

                               select E.*
                               from
                                      (
                                         select NULL C1, PROV_ID
                                         from (
                                                select PROV_ID, sum(units) suma0
                                                from DBO.VENTAS_MX_20'+@v+
                                                ' group by PROV_ID
                                              ) C
                                         where suma0 = 0
                                      ) D join (
                                                select PROV_ID as cod_VS
                                                from DBO.PROVEEDOR_MASTER
                                                where tipo = VS
                                               ) E on D.PROV_ID = E.cod_VS
                                    )  K on J.VS = K.cod_VS
                  )  N on M.PROV_ID = N.VS
           ) p
on PROV_ID = p.new_cod_VS
group by '+@columnas
)
end
else
   begin
       set @v=@v+88
       set @y=@y+88
       set @z = convert(varchar,@y)
   end
end
```

Código 7: SP_VENTA_VS

En el código 7 presento el del procedimiento almacenado *sp_venta_vs* el cual ejecuta una consulta a las ventas de los productos y puntos de venta especificados en las tablas parámetro *maestro* y *mercado* en el periodo definido entre *date1* y *date2* pero aplicando las reglas de negocio para los proveedores con venta VS explicadas en el punto anterior.

4.2.2.3 Resultados obtenidos

Gracias a la automatización de estas de reglas de negocios fue posible construir la venta VS mensualmente de forma paralela a los otros departamentos que también la generan, de esta forma, el departamento de servicio al cliente pudo dar respuestas rápidas a las peticiones de los clientes sin tener que depender de otras áreas para poder hacerlo; de igual manera, al tener programadas estas reglas en un procedimiento almacenado sirvió de soporte para otras áreas que solo fungen como usuarios de las bases datos y ocupan la información de esa venta para realizar sus propios análisis y, por lo tanto, todas las áreas pertinentes dentro de la empresa estuvieron alineadas a un solo resultado cuando se trate de este tipo de venta.

4.2.3 Validación de venta interna

Las validaciones de venta interna consistían en verificar que los datos que manejaba la empresa correspondían a los mismos que el cliente nos entregaba, de esta manera el cliente comprobaba el buen manejo de sus datos por parte nuestra. Mi tarea fue hacer las consultas necesarias de nuestros datos de acuerdo con los datos que el cliente había entregado y demostrar que ambas datas tenían una tendencia muy parecida. Validaciones de este tipo se hacían con la finalidad de que el cliente contrate los servicios de ciertas auditorias.

4.2.3.1 Antecedentes

Como lo he mencionado con anterioridad, el departamento de servicio al cliente era el lazo entre la empresa y los laboratorios, por lo tanto, cada que se planificaba la integración de un nuevo laboratorio al portafolio de clientes de la empresa, se comenzaba a desarrollar un proyecto que incluía, entre muchos otros procesos, la validación de sus datos contra los datos que la empresa maneja. De esta forma, si sus datos coincidían en un alto porcentaje con los nuestros, significaba que la calidad de la información que maneja la empresa era buena y, por consecuente, el cliente estaría propenso a contratar los servicios.

Mi función como analista de bases de datos fue, a grandes rasgos, realizar el cruce de datos de ambas fuentes y analizar la información para detectar datos erróneos.

4.2.3.2 Metodología de resolución

Para facilitar el proceso de cruce de información, envié a los clientes un formato donde se especificaba el layout con el que nos debían compartir los datos. En dicho layout quedaba definido el nivel de apertura al que se haría dicha validación.

Una vez que el cliente compartía sus datos, los limpiaba y les daba un formato legible para que la relación con nuestros datos fuera más sencilla.

Para obtener la información que nosotros manejábamos, elaboré scripts SQL con la misma estructura de los ya explicados en el Código 6, pues de igual manera definía una tabla maestro y mercado, e iba consultando las ventas de los meses requeridos.

Además, en estos scripts, de acuerdo con los requerimientos a los que se habían llegado con el cliente, definía los filtros necesarios y extraía la apertura solo de las dimensiones acordadas.

El único inconveniente que había era el tiempo de ejecución de la consulta ya que en la mayoría de estas validaciones el nivel de apertura era el máximo, además, se consultaban más de 60 meses de información.

Para visualizar la información utilicé la herramienta Microsoft Power BI, la cual es un servicio de inteligencia empresarial, que permite unir diferentes fuentes de datos, modelizar y analizar datos para después, presentarlos a través de paneles e informes. [17]

A través de esta herramienta, mediante los gráficos, logré detectar la información errónea: gráficas donde se rompía tendencia, ventas no encontradas, o productos con diferente descripción. A partir de este punto, mi tarea fue volver a consultar las bases de datos y corroborar dicho error para detectar si provenía de la fuente del cliente o de la nuestra, y en caso afirmativo, alertar dicho error.

Una vez que se corregían los errores, elaboraba los dashboards en Power BI para presentar a los clientes su validación de datos.

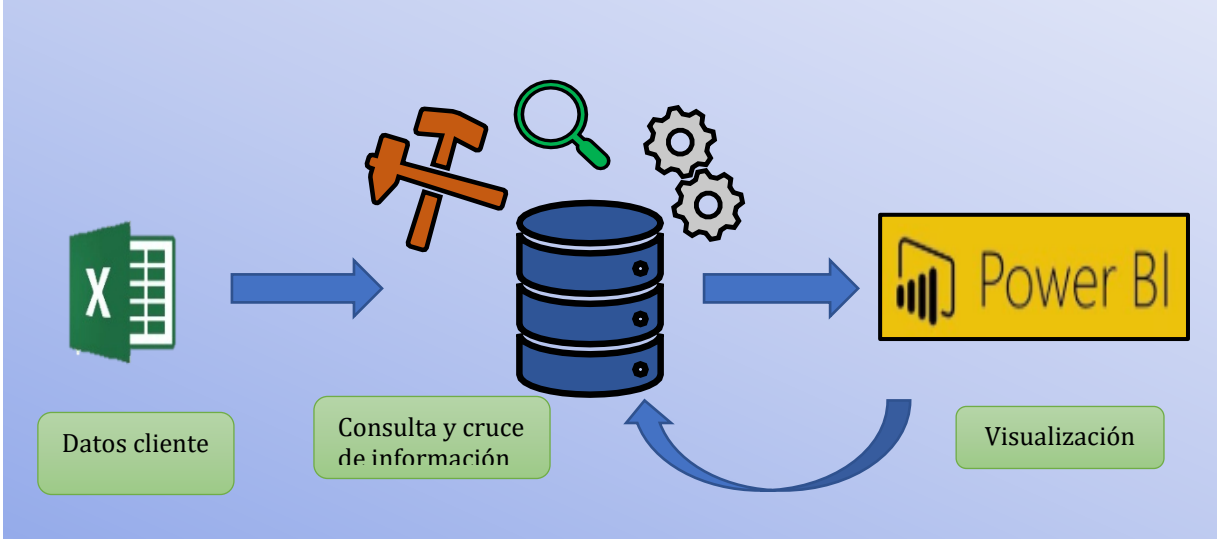


Figura 15: Proceso de Validación de Venta Interna

En la Figura 15 presento el proceso para llevar a cabo la validación de venta del cliente contra nuestras fuentes de información disponibles. Se extrae la información de hojas de cálculo enviadas por el cliente, se cargan a tablas dentro de la base de datos y se realizan las consultas y cruces necesarios para detectar la presencia de errores, y al final, desde Power BI, se visualizaba la información y se presentaba al cliente.

4.2.3.3 Resultados obtenidos

Participé en proyectos como este, cuya finalidad era que el cliente contratara los servicios. En los cuales fue posible que, durante la presentación de la validación, el mismo cliente interactuara con los dashboards presentados y pudiera ver, al nivel de apertura que el deseara, la mayor información posible permitida, logrando así, que el cliente quedara satisfecho con el manejo y la calidad de la información y contratara el servicio de la empresa.

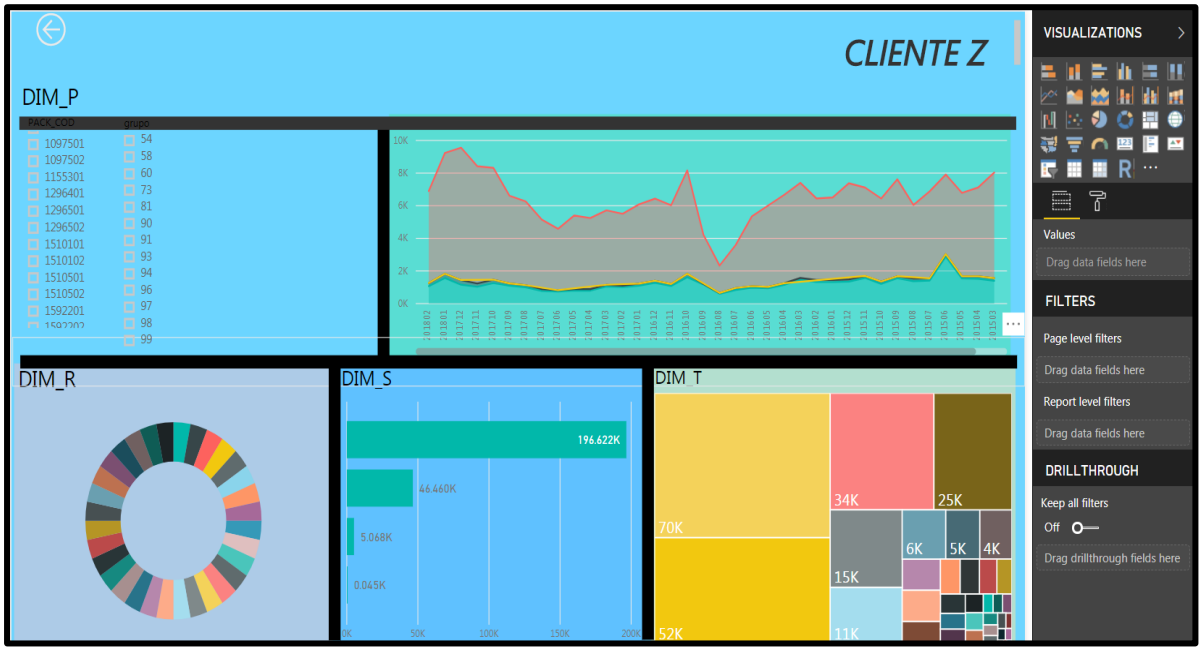


Ilustración 11: Plantilla de Dashboard en Power BI

En la Ilustración 11 muestro la plantilla que diseñé para presentar estas validaciones a clientes. En esta podían visualizar, a través de múltiples gráficos, el comportamiento de los KPI, modificando las dimensiones para obtener el detalle y apertura deseado.

Hasta este punto he explicado mi participación como analista de datos en el área de servicio al cliente, destacando mis funciones y actividades principales.
A continuación, describiré mi rol dentro del departamento de Tecnología y Aplicaciones.

4.3 Como analista de Sistemas de Negocio

4.3.1 Diseño de procesos ETL, construcción de cubos y administración de herramientas BI

Debido a proyectos de migración de datos, una de mis actividades fue la de diseñar procesos de extracción, transformación y carga de varias auditorias de información para que estas fueran visibles a través de las nuevas herramientas de BI a donde se migraron. De igual forma, construí y generé cubos de información para su carga a las plataformas BI.

4.3.1.1 Antecedentes

Debido a la modificación en la infraestructura tecnológica de la empresa, se pusieron en marcha nuevas etapas enfocadas en la mejora de la calidad de la información, con esto, se planificaron proyectos de migración de sistemas legados a nuevas herramientas y plataformas que se ajustaban de una mejor manera con las necesidades del mercado.

La migración se realizó de antiguas plataformas de inteligencia de negocios a nuevas herramientas con múltiples mejoras en cuanto al desempeño e interpretación de la información.

El reto se enfocó en modificar y diseñar procesos de extracción, transformación y carga ETL de múltiples fuentes de información a las nuevas plataformas, con la condición de que la información continuara en estado productivo, cuidando la integridad de los datos y no alterar la vista final a los usuario o clientes.

4.3.1.2 Metodología de resolución

Tuve bajo mi cargo la tarea de diseñar procesos de integración de datos para migrarlos, construir los cubos de información y cargarlos a las nuevas plataformas BI con el fin de que dichas auditorias continuaran en estado productivo.

Considero necesario mencionar algunos aspectos importantes que llevé a cabo antes de diseñar dichos procesos:

1. Me capacité en el uso de la herramienta BI a donde cargaría los cubos de información: a nivel administrador para conocer sus especificaciones técnicas y entender los modelos de datos que soporta, así como a nivel usuario para saber cómo visualizar e interpretar la información.
2. Estudié el modelo final al que se tenía que llegar y la auditoria de información al nivel del negocio para estar consciente acerca de qué es lo que el cliente necesitaría analizar y visualizar.
3. Detecté las fuentes de información para saber qué se necesitaría y de donde lo podía tomar.
4. Me orienté acerca de las modificaciones necesarias a realizar para poder integrar dichas fuentes.
5. Estudié qué información contendría cada dimensión, así como las métricas de la tabla de hechos.
6. Después de la generación de cada cubo, realicé pruebas del desempeño de la herramienta final.

Diseño de sistema ETL

A continuación, explicaré cómo realicé el diseño del proceso ETL:

- Extracción

Después de haber realizado lo mencionado en los primeros puntos de arriba, identifiqué las fuentes de información de donde obtendría los datos. Dichas fuentes venían en tres formatos distintos: archivos de texto plano, hojas de cálculo excel y archivos dbf. También cabe destacar que estos archivos se encontraban en distintas ubicaciones: carpetas en red o servidores diferentes.

El software que utilicé para el diseño de estos procesos fue SQL Server Integration Services (SSIS).

Lo primero que extraje fueron los catálogos que, posteriormente, los utilicé para la construcción de las dimensiones. En SSIS se crean conexiones a las fuentes de datos para extraerlos. En este caso, creé conexiones Excel y OLE DB; esta última fue usada pues los archivos DBF son extraídos usando este tipo de driver.

- Pruebas realizadas

En un paso previo a lo mencionado en el párrafo anterior, ejecuté múltiples pruebas para poder llegar a esa conclusión, pues en primera instancia, configuré las conexiones dentro del SSIS como conexiones Excel para la extracción de los archivos DBF, esto como consecuencia a que no existe un driver destinado a realizar la conexión con archivos en este formato, por lo tanto, abrí y guardé estos archivos desde Excel para utilizar dicha conexión.

Posteriormente, debido al alto número de archivos DBF que tenía que extraer, fue ineficiente estar convirtiéndolos a archivos Excel. Por lo tanto, después de haber investigado el tema, descubrí que es posible hacerlo mediante la conexión OLE DB del SSIS pero realizando alteraciones a la configuración default de este tipo de conexiones.

Con esto, fue posible acelerar el tiempo de extracción de los archivos fuente.

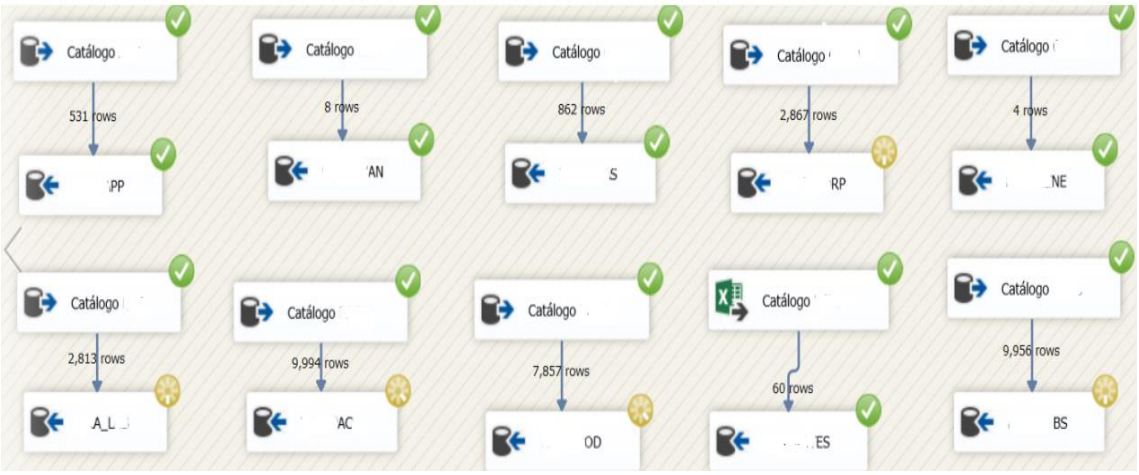


Ilustración 12: Carga de Catálogos

En la Ilustración 12 muestro la extracción de los catálogos de archivos DBFy Excel usando la herramienta SSIS.

Ya que el archivo que contenía la tabla de hechos era el más pesado, tuve que utilizar otras técnicas para realizar la carga masiva de estos datos, pues después de haber realizado pruebas cargando este tipo de tablas con el SSIS todas consumían muchos recursos y la carga era lenta.

Así que ejecuté la carga desde SQL:

```
bulk insert [APPLCATION_BI].[RAW_DATA]
from '\\SERVERTYUIOP34\APP_BI_IN\CLIE\FACT_012_MX.txt'
with
(
rowterminator ='\n'
)
```

Código 8: Bulk Insert

El código 8 es un fragmento SQL donde utilizo bulk insert, el cual es una sentencia que ejecuta la importación masiva de archivos de texto plano y los carga a una tabla previamente creada dentro de una base de datos.

- Transformación

Una vez que tenía la tabla de hechos almacenada en la base de datos, lo siguiente que hice fue darle el formato requerido para que ésta pudiera ser leída y apta para la construcción del cubo, ya que se trataba de datos en crudo, es decir, sin ninguna transformación aplicada. Por formato me refiero a aplicar la separación de columnas de acuerdo con la longitud definida de cada una.

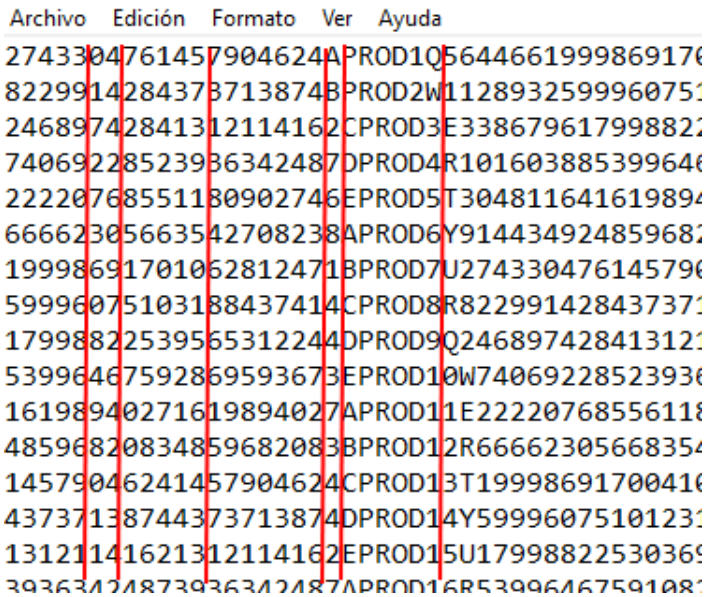


Ilustración 13: Data Cruda

La Ilustración 13 es un ejemplo del formato de la data de los archivos fuente. Como se observa, son solo caracteres que no nos aportan información y es necesario ajustarlos a un formato para poder interpretarlos. Al tenerlo ya almacenado en una tabla de una base de datos aún es necesario realizar la separación de caracteres para que se puedan formar las columnas definidas.

Dado que la tabla de hechos se encontraba desnormalizada, es decir, las descripciones de la dimensión tiempo se encontraban como columnas, una columna por cada periodo de tiempo, la tabla horizontalmente era muy grande.

De lo anterior surge la necesidad de automatizar la generación de columnas en la tabla de hechos de acuerdo con su longitud dada, pues manualmente sería un proceso tardado, poco eficiente y propenso a errores. Es por esto que programé un procedimiento almacenado que contiene ya la información del mapeo y longitud de columnas para la tabla de hechos de esta auditoria específicamente.

```
create proc sp_deriva_columas
@fecha_input_str varchar(10), @length_input varchar(5), -- Parámetros de entrada
@tabla_destino varchar(40)
as
begin
declare @w date, @año varchar(10), @mes varchar(10)
declare @count int, @decre int, @start int, @start_l varchar(5)
set @count = 1 -- Cuenta el número de ciclos
set @decre = 1 -- Decrementa el mes
set @start = 10 -- Define el número de caracter de INICIO de cada columna <UNITS>
print
'select SUBSTRING([Column 0],1,9) CLIE, SUBSTRING([Column 0],10,1) GTP,
CONVERT(INT,SUBSTRING([Column 0],11,10)) ZON, SUBSTRING([Column 0],21,4) COD,'
while 24 >= @count -- El ler ciclo se repetirá 24 veces
     begin
          set @decre = @decre-1 -- A partir de la fecha dada como cadena, decrementa 1 unidad
          set @count = @count+1 -- Aumenta una unidad hasta que se cumpla el número de ciclos
          set @start = @start + 15 -- La primera columna de <UNITS> comienza en la posición 25
          set @start_l= convert(varchar,@start)
          set @w = dateadd(month,@decre,@fecha_input_str) -- Dada la fecha, realiza el decremento mensual
          set @año = substring(convert(varchar,@w),1,4) -- Del formato generado, toma el año
          set @mes = substring(convert(varchar,@w),6,2) -- Del formato generado, toma el mes
          print +'CONVERT(INT,SUBSTRING([Column 0],'+@start_l+','+@length_input+')) UNITS_'+@año+@mes+','
     end -- Termina ler ciclo
-- Reiniciar valores de viariables contadoras
set @count = 1
set @decre = 1
set @start = 370 -- Define el número de caracter de INICIO de cada columna <UNITS>
while 24 >= @count
     begin
          set @decre = @decre-1
          set @count = @count+1
          set @start = @start + 15 -- La primera columna de <VALUES> comienza en la posición YYY
          set @start_l= convert(varchar,@start)
          set @w = dateadd(month,@decre,@fecha_input_str)
          set @año = substring(convert(varchar,@w),1,4)
          set @mes = substring(convert(varchar,@w),6,2) -- Concatena con la salida del ler ciclo v
          print +'CONVERT(FLOAT,SUBSTRING([Column 0],'+@start_l+','+@length_input+')) VALUES_'+@año+@mes+'
     end -- Termina 2do ciclo
print +'CONVERT(INT,SUBSTRING([Column 0],745,25)) SumControl_Units_'+@fecha_input_str+',
CONVERT(FLOAT,SUBSTRING([Column 0],770,25)) SumControl_Values_'+@fecha_input_str+'
INTO '+@tabla_destino+' from [RAW_DATA]'
end
```

Código 9: SP_DERIVA_COLUMNAS

En el Código 9 muestro el procedimiento almacenado que programé, donde recibía como parámetros de entrada el periodo y la tabla destino. A partir del periodo dado recorría hacia atrás todos los periodos de la dimensión tiempo y realizaba la derivación de columnas de acuerdo con la longitud de cada una, de igual forma, les generaba los nombres a todas las columnas y las almacenaba en la tabla destino.

La salida de este procedimiento no era como tal la ejecución de la tarea, sino que era la generación de una cadena de texto que contenía la consulta que realizaba dicha tarea, esto lo pensé de esta forma para validar antes que la consulta que estaría a punto de ejecutarse fuera correcta.

Con la finalidad de que este sub proceso estuviera lo más automatizado posible, ejecuté este procedimiento almacenado usando sqlcmd, utilidad que permite ejecutar scripts y sentencias T-SQL desde línea de comandos. [18]

```
sqlcmd -S SERVERTYUIOP34 -d APPLICATION_BI -E -Q "exec sp_deriva_columnas
'20181201','15','DBO.COLUMNS_DERIVED'" -o "Docuementos\STRING_QUERY.sql" -s","
```

Código 10: Sentencia CONNECT_BD_EXEC_SP en SQLCMD

El uso de SQLCMD se ejemplifica en el Código 10.

La cadena resultado que contenía la consulta era guardada en *STRING_QUERY.sql*, el cual era llamado desde una tarea en SSIS desde donde se ejecutaba y realizaba las modificaciones en la base de datos.

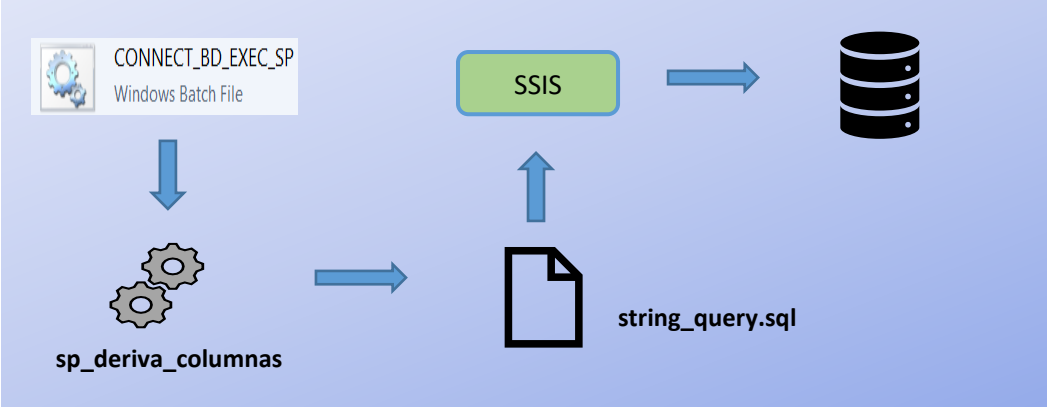


Figura 16: Flujo de Ejecutables

La Figura 16 describe el flujo de los ejecutables y los archivos y acciones generados de cada uno.

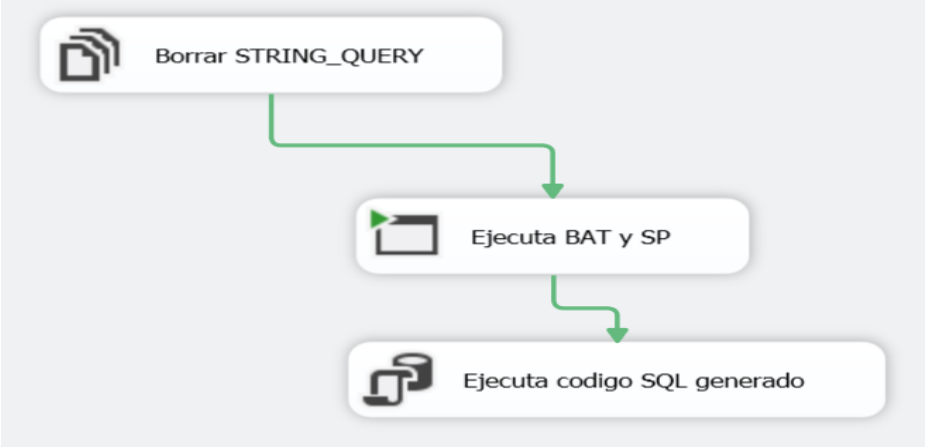


Ilustración 14: Transformación a Tabla de Hechos con SSIS

En la Ilustración 14 muestro el flujo de los procesos mencionados arriba pero ya usando SSIS. Cada caja es una tarea en donde se ejecutaban los archivos bat y sql.

Una vez que la tabla de hechos se encontraba correctamente cargada y con un formato adecuado, lo siguiente que hice fue crear las dimensiones que contendrían los cubos. Para poder construirlas utilicé los catálogos ya cargados.

Las trasformaciones aplicadas variaban en cada dimensión, pero de forma general, realicé mapeos de columnas de origen a destino, joins entre tablas, cálculos aritméticos y normalización de tablas.

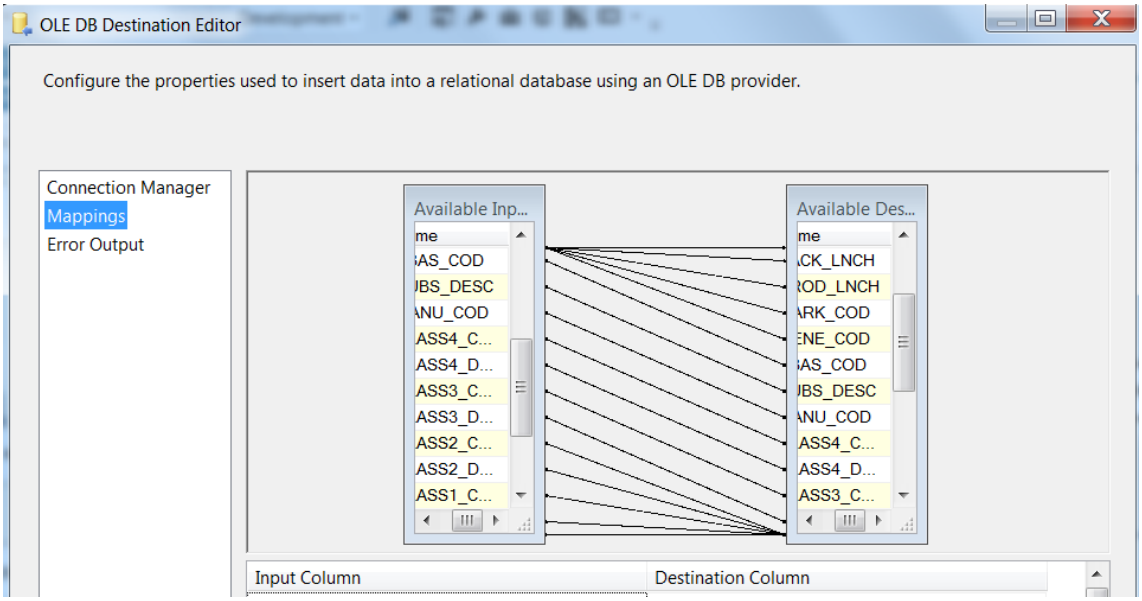


Ilustración 15: Mapeo de Columnas

El mapeo de columnas se muestra en la Ilustración 15. Lo realicé para definir a qué columnas se cargarían los datos de una tabla origen a una tabla destino.

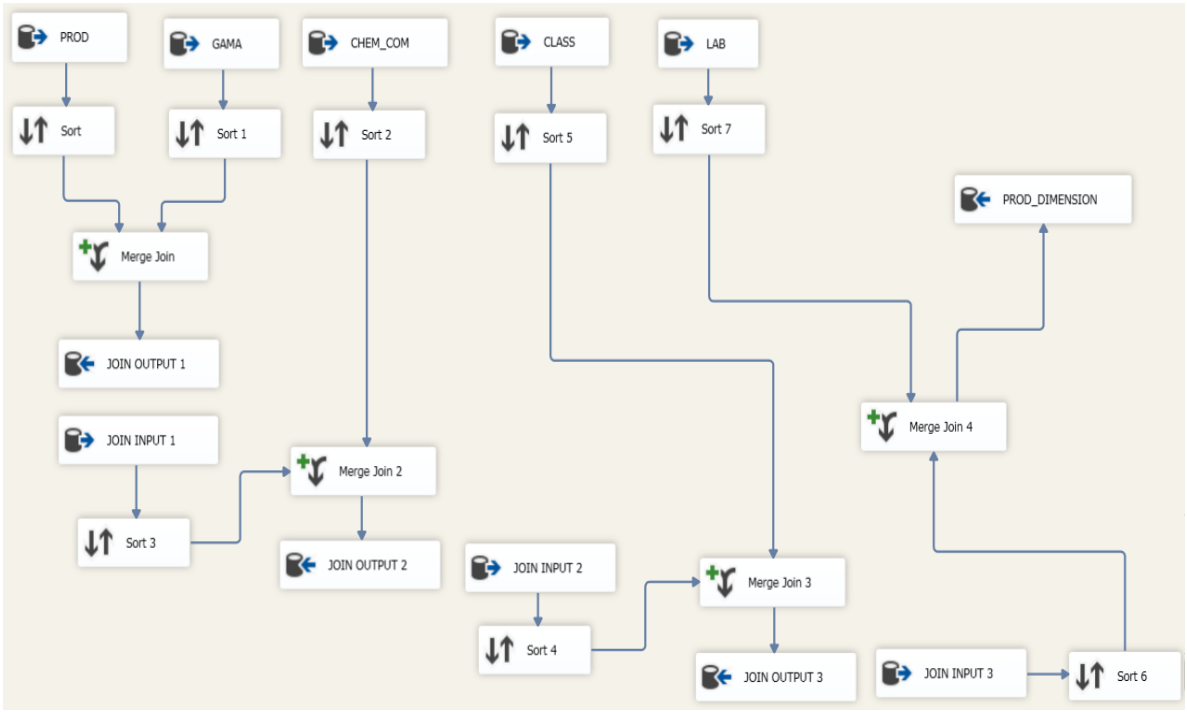


Ilustración 16: Joins en SSIS

En SSIS es posible realizar joins entre tablas para facilitar las tareas, como lo muestro en la Ilustración 16, pero fue necesario cubrir ciertos aspectos para que estos funcionaran correctamente.

Enter SQL Query

```
select right(a.periodo,8) as [DATE], a.PROD, a.CHANNEL, a.UNITS as UNITS, b.PRICE1, c.PRICE2
into dbo.Pre_Fact
from DBO.D_UNITS a join DBO.D_PRICE_1 b
on concat(right(a.periodo,6),a.channel,a.PROD) = concat(right(b.periodo,6),b.channel,b.PROD)
join DBO.PRICE_2 c
on concat(right(b.periodo,6),b.channel,b.PROD) = concat(right(c.periodo,6),c.channel,c.PROD)
-------------------------------------------------------------------------------------------------------------------
select [Date], Prod, Channel, sum(convert(int,UNITS)) as UNITS, sum(convert(float,PRICE1)) as PRICE1, sum(convert(float,PRICE2)) as PRICE2
case when Channel in ('A','G','K','U') then convert(float,PRICE1) else convert(float,PRICE2) end QQQ,
case when Channel in ('R','Y','O','W','H','E','D') then DINAM*(convert(float,PRICE1)) else DINAM*(convert(float,PRICE2)) end NNN -- Funciona
into DBO.FACT_TABLE -- FINAL
from dbo.Pre_Fact a join (select FECHA, 1/rate*1000 as DINAM
                                                    from [dbo].[RATES]
                                                    ) b
on a.[Date] = b.Fecha
group by [Date], Prod, Channel, PRICE1, PRICE2, b.DINAM
```

Enter SQL Query

```
select [Date], Channel, Gama, Prod, Port, Periodo, UNITS
into dbo.UNPIVOT_UNITS
from dbo.D_UNITS
UNPIVOT
    (
        UNITS
        for [Periodo] in (UNITS_20140301,    UNITS_20140401,    UNITS_20140501,    UNITS_20140601,
                          UNITS_20150401,    UNITS_20150501,    UNITS_20150601,    UNITS_20150701,
                          UNITS_20160501,    UNITS_20160601,    UNITS_20160701,    UNITS_20160801,
                          UNITS_20170501,    UNITS_20170601,    UNITS_20170701,    UNITS_20170801,
                          UNITS_20180401,    UNITS_20180501,    UNITS_20180601,    UNITS_20180701,
    ) as P
```

Ilustración 17: T-SQL en SSIS

Como se muestra en la Ilustración 17, el proceso que tuve que diseñar se tornó más complejo, por lo tanto, ejecuté sentencias T-SQL desde SSIS. En este caso, en la primera parte de código, efectué joins entre varias tablas, añadí condicionales para modificar tipos de datos, y se realicé cálculos aritméticos. En la segunda parte, realicé la transposición de la tabla para que ésta quedara normalizada, obteniendo así, la dimensión de tiempo en una sola columna.

En este punto, tanto la tabla de hechos como las tablas de dimensiones se encontraban listas para ser cargadas a la herramienta BI y efectuar la construcción de cubo posteriormente.

Ilustración 18: Sistema ETL en SSIS

En la Ilustración 18 muestro el flujo de datos en SSIS de los sistemas ETL que diseñé. Como lo expliqué a lo largo de esta sección, dicho sistema extrae la información de diferentes fuentes y sitios, utiliza diversas técnicas para la carga de datos de acuerdo con el tamaño de los archivos, realiza las trasformaciones necesarias para que la información pueda ser correctamente almacenada en tablas y construye las dimensiones del cubo llevando a cabo cruces y transformación de tablas, así como diversos cálculos aritméticos.

- Carga

Todo lo concerniente a la carga de información a la herramienta BI lo llevé a cabo desde la herramienta misma. Como lo mencioné en los puntos al inicio de esta sección, fue necesario capacitarme en el uso y administración de ésta para poder efectuar la carga y construcción de los cubos.

La herramienta es muy intuitiva y con una interfaz amigable. Los archivos de hechos y dimensiones eran cargados a un servidor de donde eran tomados por la herramienta.

Dicha herramienta estaba construida para recibir una tabla de hechos y múltiples catálogos los cuales interpretaba como dimensiones. Estaba apta para interpretar

modelos de datos tipo estrella. Una vez con los archivos cargados, se realizaban los joins correspondientes de todas las dimensiones con la tabla de hechos para la construcción del cubo. La herramienta validaba las llaves primarias y foráneas de cada archivo.

Una vez que el cubo se encontraba listo, realizaba validaciones dentro de la herramienta para que pudiera ser generado y publicado a los usuarios finales, es decir, los clientes.

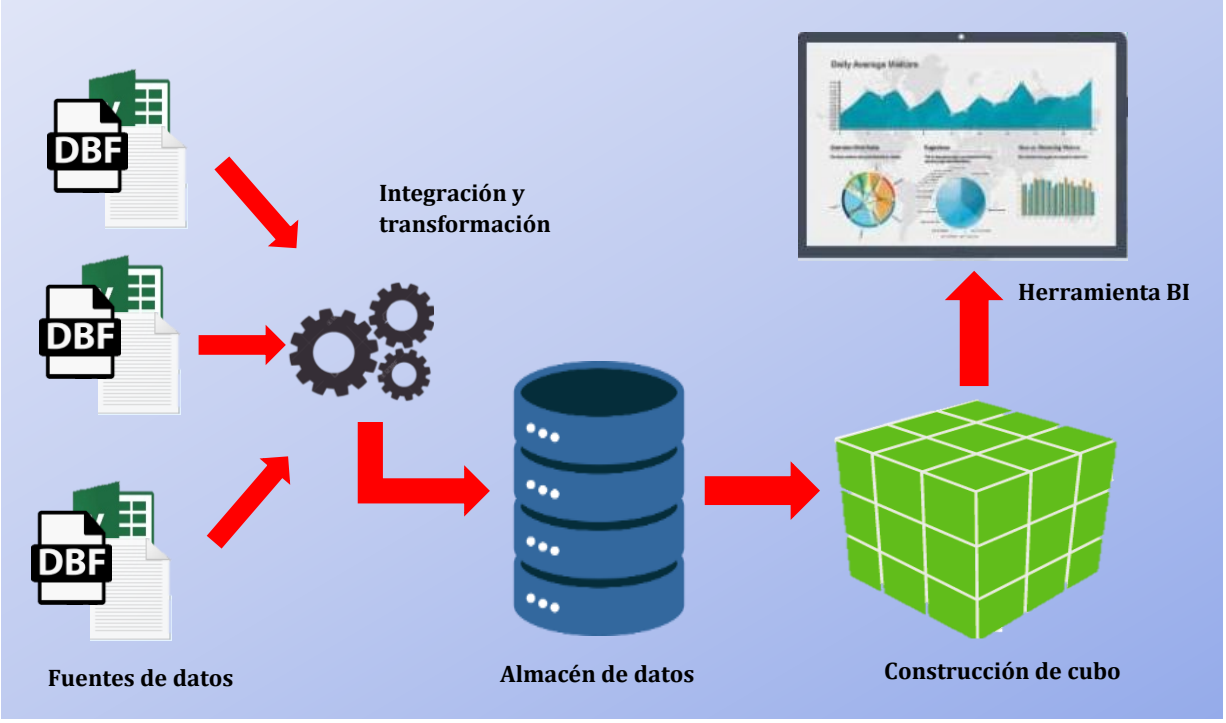


Figura 17: Sistema ETL

La Figura 17 muestra el proceso general desde el sistema ETL que diseñé hasta la construcción y carga de cubos a la herramienta BI.

4.3.1.3 Resultados obtenidos

La migración de esta auditoria no presentó ningún percance pues se conservó la integridad de los datos y siempre se mantuvieron en estado operativo.

Yo realicé todo lo mencionado en esta sección para 26 cubos de información – uno por cliente – y se convirtió en mi responsabilidad para ser llevada a cabo mensualmente. Posterior a la liberación a clientes, di asesoramiento y capacitación a los mismos acerca del uso y desempeño de la herramienta.

Los clientes no tuvieron problema alguno para visualizar y analizar la información; la migración de esta auditoria se dio como exitosa pues se ejecutó en tiempo y forma para los clientes con los que se tenía pactado en el contrato, es decir, para los que ya habían pagado dicho servicio.

Los procesos ETL que diseñé fueron productivos para la integración de otras auditorias que faltaba migrar.

Hasta este punto concluye mi quehacer en este departamento, así como también cierro el desarrollo de las principales actividades dentro de los departamentos a los que pertenecí en mi trayectoria laboral descrita al inicio.

En el siguiente capítulo, presentaré mis conclusiones generales que cerrarán este informe.

7. Conclusiones

Gracias al auge de las tecnologías de información la cantidad de datos que se generan a diario en empresas de cualquier giro continúa creciendo, y consecuentemente, como lo desarrollé en este informe, la necesidad de analizar dichos datos y convertirlos en información de valor también crece.

La demanda de profesionistas en México expertos en temas de análisis de información es muy alta; tendencias tales como Inteligencia de Negocios, Big Data y Ciencia de Datos, enfocadas en el análisis y generación de conocimiento continúan posicionándose dentro de los perfiles más solicitados en el mercado laboral.

Sin duda alguna, el plan de estudios de la carrera de Ingeniería en Computación de la UNAM me preparó para afrontar estos retos y demandas del mercado. Los conocimientos que adquirí a lo largo de la carrera me sirvieron como base para entender y buscar la solución más optima a los problemas que iba encontrando dentro de mis funciones diarias en los departamentos que laboré, demostrando así, que cumplo con los requerimientos necesarios para desempeñar y afrontar satisfactoriamente las actividades para las que fui contratado.

En todos los quehaceres en los que me desenvolví explicados en este informe, apliqué los fundamentos teóricos que adquirí en algún punto de mi trayectoria escolar, todas las labores y responsabilidades que me fueron asignadas las cumplí en tiempo y forma con base en los objetivos fijados en cada uno.

No solo fueron los conocimientos aprendidos directamente dentro de las aulas, sino también fueron los buenos hábitos que la Facultad de Ingeniería mi hizo adquirir a lo largo de la carrera, como aprender a administrar mejor el tiempo, saber priorizar las cosas, o ser autodidacta, fue lo que me ayudó a tener un mejor desenvolvimiento al ejercer todo lo que he estudiado.

Más allá de aplicar los conocimientos propios de la Ingeniería en Computación, desarrollar las llamadas *soft skills* fue un reto igual o mayor, pues cuestiones como entender y dominar el lenguaje del negocio, tratar con un cliente o expresar adecuadamente mis ideas en una reunión fueron igual de desafiantes, por lo tanto, significó un doble reto dominar ambos mundos, en los cuales mi desarrollo aún no termina.

Cabe destacar que el camino todavía es largo, como lo mencioné, lo aprendido durante la carrera solo fueron las bases del conocimiento, sirvieron como una introducción y me permitieron insertarme exitosamente dentro del campo laboral para enfrentarme a proyectos reales que demanda la industria de tecnologías de la información, por mi parte, tengo el deber de seguir capacitándome, actualizándome, continuar aprendiendo y desarrollando nuevas habilidades para atender las demandas y necesidades del mercado laboral ejerciendo como Ingeniero en Computación.

8. Referencias

[1] Oracle, «Oracle,» [En línea]. Available: https://www.oracle.com/ocom/groups/public/@otn/documents/webcontent/317529_esa.pdf. [Último acceso: 19 Abril 2019].

[2] IBM Software Group, «IBM,» [En línea]. Available: https://www-07.ibm.com/sg/events/blueprint/pdf/day1/Introduction_to_Business_Intelligence.pdf. [Último acceso: 14 Abril 2019].

[3] LatinoBI, «LatinoBI Inteligencia de Negocios + Soluciones Estratégicas,» 2013. [En línea]. Available: https://www.latino-bi.com/espanol/fundamentos-bi/introduccion-data-warehouse.php. [Último acceso: 14 Abril 2019].

[4] id00710310, «Diarium,» Univesidad de Salamanca , 16 Marzo 2016. [En línea]. Available: https://diarium.usal.es/id00710310/2016/03/16/business-intelligence/. [Último acceso: 2019 Abril 14].

[5] Mendez, A., Mártire, A., Britos, P. Y Garcia-Martínez, R, «Fundamentos de Data Warehouse,» Reportes Técnicos en Ingeniería del Software, vol. 5, nº 1, pp. 19-26, 2003.

[6] DataPrix, «DataPrix: Knowledge Is The Goal,» [En línea]. Available: http://www.dataprix.com/qu-es-un-data-warehouse. [Último acceso: 20 Abril 2019].

[7] A. C. Trujillo, «Modelo Multidimensional,» Ingenieria Industrial, vol. XXVII, nº 1, pp. 15-18, 2006.

[8] A. F. Morales, R. E. C. Valencia y J. M. M. Castro, «Procesamiento Analítico con Mineria de Datos,» Revista Iberoamericana de las Ciencias Computacionales e Informática, vol. 5, nº 9, 2015.

[9] División Consultoría de EvaluandoSoftware.com, «Evaluando Software,» 16 Agosto 2016. [En línea]. Available: https://www.evaluandosoftware.com/cubos-olap-informacion-la-toma-decisiones/. [Último acceso: 2019 Abril 18].

[10] SAS, «SAS,» [En línea]. Available: https://www.sas.com/es_mx/insights/data-management/what-is-etl.html. [Último acceso: 18 Abril 2019].

[11] M. T. Özsu, «Fragmentation,» de Principles of Distributed Database Systems, Nueva York, Springer, 2011, pp. 81-83.

[12] CVA, «Centro Virtual de Aprendizaje,» [En línea]. Available: http://www.cca.org.mx/cca/cursos/estadistica/html/m14/coef_pearson.htm. [Último acceso: 14 Abril 2019].

[13] C. K. Taylor, «ThoughtCO,» 27 Abril 2018. [En línea]. Available: https://www.thoughtco.com/what-is-the-interquartile-range-rule-3126244. [Último acceso: 2019 Abril 14].

[14] Universo Fórmulas, «Universo Fórmulas,» [En línea]. Available: https://www.universoformulas.com/estadistica/descriptiva/rango-intercuartilico/. [Último acceso: 14 Abril 2019].

[15] Lumen, «Introduction to Statistics,» [En línea]. Available: https://courses.lumenlearning.com/odessa-introstats1-1/chapter/measures-of-the-location-of-the-data/. [Último acceso: 14 Abril 2019].

[16] SQL Server Tutorial, «SQL Server Tutorial,» [En línea]. Available: www.sqlservertutorial.net/sql-server-stored-procedures/. [Último acceso: 24 Abril 2019].

[17] MakeSoft, «MakeSoft,» [En línea]. Available: https://www.makesoft.es/es/que-es-power-bi/. [Último acceso: 14 Abril 2019].

[18] D. Calbimonte, «SQLShack,» 18 Octubre 2017. [En línea]. Available: https://www.sqlshack.com/working-sql-server-command-line-sqlcmd/. [Último acceso: 2019 Abril 19].