

**UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA
DEPARTAMENTO DE INFORMÁTICA
SANTIAGO - CHILE**



**Diseño e Implementación de una Solución
de Inteligencia de Negocios para la Área de
Marketing de una Multinacional Tabacalera**

Diego Andrés Lucero Marécaux

**Memoria de titulación para optar al título de
Ingeniero en Ejecución Informática**

**Profesor Guía:
José Luis Martí Lara**

Santiago, Noviembre de 2018



Resumen

En el presente documento se detalla la implementación de una herramienta BI, dando a conocer una pequeña definición de la organización, junto a un estudio de la situación actual para poder comprender los problemas principales y de esta forma definir los objetivos a cumplir. El objetivo principal es otorgar un sistema de apoyo a la gestión a una de las áreas de marketing de una empresa de la industria tabacalera. Luego de indagar en los objetivos, se especifica un marco teórico para proseguir con la solución propuesta, en la cual se detalla una descripción de la herramienta BI construida, metodología de trabajo, modelo de datos, diseño ETL e interfaz de usuario. Se termina con las conclusiones, dentro de las cuales se discute la importancia de estas herramientas dentro las organizaciones actuales.



Abstract

In this document detailed the implementation of a BI tool, giving a small definition of the organization, along with a study of the current situation to understand the major issues and in this way define the targets to meet. The main objective is to provide a support system for the management to one of the areas of marketing of a company in the tobacco industry. After a look at the objectives, its specify a theoretical framework to proceed with the solution, in which is detailed a description of built the BI tool, work methodology, data model, design ETL and interface. Ending with the conclusions, which indicate the importance of these tools within the organizations of today.



Índice General

Capítulo 1: Situación Actual	10
1.1 Definición de la organización	10
1.2 Estudio de la Situación Actual.....	11
1.3 Principales Problemas.....	12
1.4 Stakeholders.....	13
1.5 Objetivos.....	14
1.5.1 Objetivo General.....	14
1.5.2 Objetivos Específicos.....	15
1.5.3 Alcance	16
1.5.3.1Proceso de control histórico	16
1.5.3.2Proceso de reporte.....	16
Capítulo 2: Marco Teórico	18
2.1 Definición de Business Intelligence.....	18
2.2 Herramientas BI.....	22
2.3 Metodologías para un Data Warehouse	22
2.4 Modelado de datos multidimensional	25
2.5 OLAP.....	28
2.6 ETL.....	33
2.7 Modelo de ciclo de vida.....	36
2.8 Heurísticas de Nielsen	41
Capítulo 3: Solución Propuesta.....	44
3.1 Descripción de la herramienta BI	44
3.2 Elección de la metodología de trabajo.....	45
3.3 Diseño del modelo de datos	46
3.4 Diseño del ETL.....	48
3.5 Diseño de la interfaz de usuario.....	50
3.5.1 Interfaz de usuario de “MDB Information”	50



3.5.2 Interfaz de usuario de “MDB Administration”	53
3.6 Indicador clave del desempeño de la herramienta.....	55
Capítulo 4: Validación de la Solución	58
4.1. Test de Aceptación	58
4.1.1. Primera sesión	58
4.1.2. Primer reporte de errores	59
4.1.3. Segunda sesión	61
4.1.4. Segundo reporte de errores	61
Capítulo 5: Conclusiones	63
Bibliografía	67
Anexo.....	69



Índices de gráficos, tablas y figuras

Grafico 1: Funcionalidad vs. Stakeholders (4).....	21
Tabla 1: MOLAP vs. ROLAP (16)	32
Figura 1: Diagrama comparativo entre las metodologías de Inmon vs. Kimball (7)	25
Figura 2: Ejemplo de Modelo de datos Estrella (9).....	26
Figura 3: Ejemplo de Modelo Copo de Nieve (9)	27
Figura 4: Ejemplo de Modelo Copo de Estrellas (9).....	28
Figura 5: Ejemplo de arquitectura MOLAP (14)	30
Figura 6: Ejemplo de arquitectura ROLAP (14)	31
Figura 7: Ejemplo de arquitectura HOLAP (14)	33
Figura 8: Modelo del ciclo de vida en cascada (20).....	37
Figura 9: Modelo del ciclo de vida en V (20)	38
Figura 10: Modelo del ciclo de vida iterativo (20).....	39
Figura 11: Modelo del ciclo de vida incremental (20)	40
Figura 12: Diagrama de la Base de Datos	47
Figura 13: Menú Principal.....	51
Figura 14: Ejemplo de Reporte Automatizado de la participación de mercado por marca	52
Figura 15: Ejemplo de ventana de una selección de acción agregar SKU	54
A - 1: Ejemplo de Hoja con Tabla Dinámica	69
A - 2: Menú MDB Administration	69
A - 3: Template de Carga	70
A - 4: Ejemplo ventaja sub-menu.....	70



Introducción

En la actualidad, las empresas suelen limitarse a almacenar información histórica, otorgando gran importancia a los repositorios de datos (*data warehouse*). El valor de estos repositorios radica en que mientras más robusta sea la información y mejor organizada esté, más fácil será realizar consultas complejas a las áreas comerciales.

Las nuevas tecnologías han demostrado una insuperable habilidad para facilitar las consultas en línea de información histórica. Para esto, las herramientas no son pocas; desde una simple tabla dinámica en Excel hasta los más estructurados cubos OLAP (*OnLine Analytical Processing*). No obstante, grandes empresas aún tienden a conformarse con almacenar información dentro de carpetas en servidores *online*, aunque, también hay empresas que crean cubos de consultas para que los usuarios se "auto-atendan" y analicen su propia información histórica. Eventualmente, algunas organizaciones han creado departamentos de inteligencia de mercados, los que atienden análisis de información a pedido, además de administrar los grandes repositorios de datos.

Dentro de los procesos del Departamento de Marketing Southern Cone de la empresa multinacional British American Tobacco, es necesario mejorar el control, el seguimiento y la verificación de la información que posee, y que se entrega a distintas áreas. Una herramienta de inteligencia de negocios (BI, del inglés *Business Intelligence*) permitirá mejorar la calidad de la información, evitando archivos duplicados y optimizando el desempeño de las áreas. Considerando la magnitud de la empresa, una herramienta BI se dispone a implementarse en un área específica, que corresponde al



área de SPI (*Strategy, Planning & Inside*), la cual se encuentra a cargo del análisis de los mercados de Argentina, Bolivia, Chile, Paraguay, Perú y Uruguay.

Los procesos de análisis del área de SPI se basan, principalmente, en los datos recopilados de diversas fuentes externas e internas. Además, se debe considerar los plazos de entrega de reportes en donde la información verídica tiene un rol fundamental. Desafortunadamente lo reportado por el área se réplica en archivos Excel, y en caso de haber un cambio no se puede hacer seguimiento de toda la gente que requirió de la información.

El presente trabajo pretende solucionar, mediante una herramienta BI personalizada en el área de SPI, la mayoría de los problemas detectados, como la falta de una única fuente oficial de información emitida por el área; también, tener un control de los usuarios de tal forma que se haga un seguimiento de quién y cuándo consultó la información, mejorar los procesos de ejecución para otorgar mayor tiempo de análisis de la información, y mantener un historial de la información, entre otros problemas que se expondrán con mayor detalle más adelante.

El informe se estructura de la siguiente forma: el capítulo 1 presenta la situación actual en la cual se encuentra la empresa donde se efectuará el trabajo, dando a conocer una pequeña definición de la organización, seguida del estado actual del área SPI, para continuar con los problemas detectados y la definición de los objetivos del trabajo. El capítulo 2 consiste en una introducción en los conceptos necesarios para comprender el desarrollo de la herramienta BI y su importancia para la empresa; se explica el significado de *Business Intelligence* y lo básico relacionado con el desarrollo de una



herramienta BI. El capítulo 3 muestra el desarrollo de la solución propuesta, donde se describe la herramienta creada, con las metodologías de trabajo implementadas, el diseño del modelo de datos, ETL y la interfaz. El capítulo 4 trata de la validación de la herramienta BI por los usuarios del área de marketing, quienes a través de pruebas emitirán una aprobación de la herramienta y/o los errores que se deben corregir. Por último, se finaliza con las conclusiones más relevantes en el desarrollo de este trabajo.



Capítulo 1: Situación Actual

En este capítulo se presenta la empresa en la cual fue implementado el desarrollo de la herramienta BI; se comienza contextualizando con una definición de la empresa para así adentrarse en la situación actual en la que se encontraba, resaltando los principales problemas detectados, junto al tipo de usuario que necesita el *software*, para terminar planteando los objetivos a cumplir con este proyecto.

1.1 Definición de la organización

El proyecto se implementó en la empresa multinacional British American Tobacco, en la área de SPI del Departamento de Marketing Southern Cone.

British American Tobacco Chile es una compañía que cuenta con una historia de más de 100 años, presente en más de 55 mercados en todo el mundo y con una figura empresarial en Chile desde 1909 (1).

Misión

“Somos una compañía tabacalera responsable que trabaja con pasión para superar las expectativas de los consumidores, clientes y accionistas, a través de la excelencia de nuestro talento y marcas ganadoras” (2).

Visión

“Nuestra visión como grupo es liderar la industria del tabaco a nivel global. Esto no lo medimos únicamente a través de volumen y valor, sino también buscamos ser los mejores en atender las necesidades de los consumidores” (2).



1.2 Estudio de la Situación Actual

La modalidad actual del área de SPI, consiste en un ciclo mensual, el cual se puede resumir en las siguientes etapas mensuales:

a) *Retail Audit*

En esta etapa se efectúan los análisis de la información recibida por empresas externas, sobre los movimientos de la venta del producto en la calle, tanto de la empresa como de la competencia. Esta información, en el minuto que es recibida, debe ser analizada, emitiendo reportes los cuales se comparten vía *mail*; en el caso de ser del mismo departamento los archivos se guardan en carpetas comunes clasificadas por fecha.

b) **SO&P**

Es la etapa más importante del Departamento de Marketing a nivel latinoamericano en la cual se presentan varios temas. En esta presentación se unen las diferentes áreas donde se analizan los aspectos positivos y negativos, los posibles problemas y las oportunidades, entre otros.

c) *Demand*

Radica en una etapa donde se analizan las proyecciones y los movimientos de la competencia de los distintos mercados, basándose en la información obtenida de *Retail Audit* y las conclusiones de SO&P.

Estas son las etapas por las cuales pasa el área SPI, dejando fuera todo tipo de reporte o requisito que pueda existir entre medio de ellas. Es por esto que también se puede apreciar la importancia de la veracidad de los datos, así como lo fundamental de tener un buen orden histórico para los análisis, ya que el tiempo apremia y no es posible desperdiciarlo en chequeos y rechequeos de la información. El Departamento de Marketing se basa en un sistema manual de varias carpetas y varios archivos Excel con tablas dinámicas sin tener ningún respaldo.

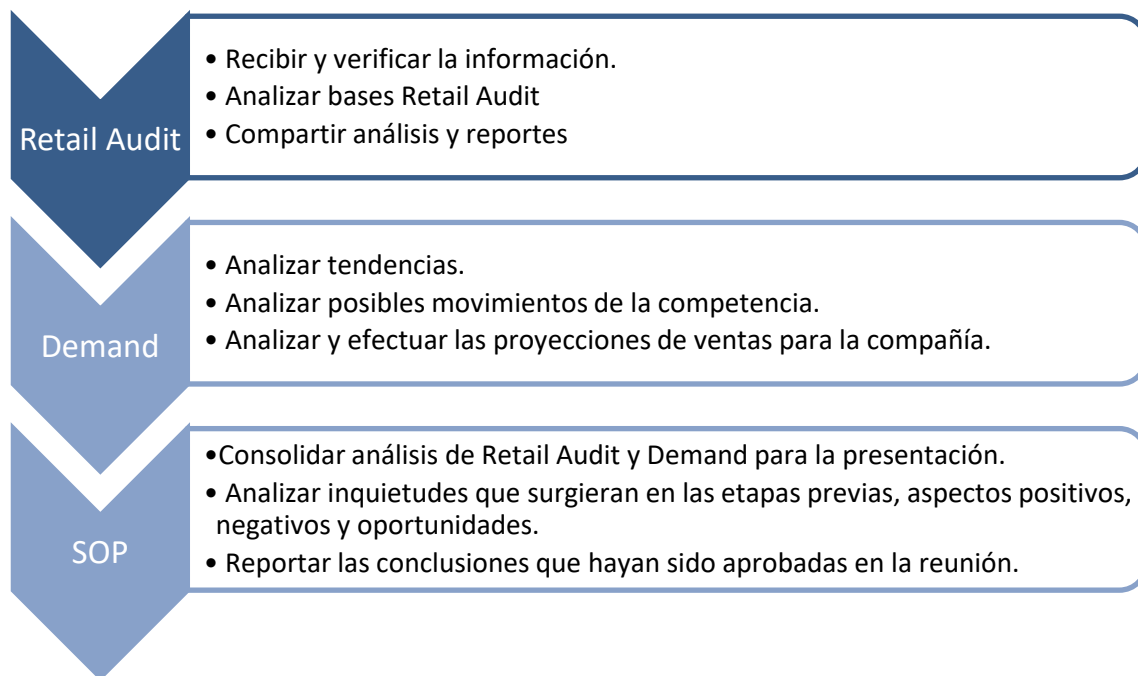


Figura 1 Diagrama de flujo de las etapas mensuales

1.3 Principales Problemas

El área no cuenta con un sistema adecuado y centralizado con el cual se pueda tener libre acceso a los datos históricos y una actualización periódica de estos. Hay una mala organización de los datos mediante carpetas y archivos individuales para cada análisis.



No existe una correcta distribución de la información, respaldándose solo en correos electrónicos con archivos adjuntos o a través de carpetas comunes, indicando en un mensaje electrónico cuál sería la ruta en donde se podría encontrar la información solicitada.

Generar reportes de análisis, en los cuales es necesario recopilar muchos datos dispersos entre diversos archivos, se traduce que sea un proceso lento y complejo.

No hay manera de saber si la información se encuentra actualizada o si hubo ajustes de algún tipo, ya que no existe un canal informativo el cual comunique a todos cada vez que los archivos sufran alguna modificación.

Finalmente, puede existir duplicidad de la información, la cual puede replicarse erróneamente en caso de haber algún ajuste.

1.4 Stakeholders

Se identifican los siguientes *stakeholders* asociados con el sistema llamado MDB (*Marketing Data Base*), donde cada tipo de usuario tiene opciones limitadas de visualización, separadas por áreas del departamento, las cuales requieren de información cruzada.

- a) **Marketing Brand:** los usuarios de la área Marcas de Marketing, encargados de la publicidad y el seguimiento puntual de las marcas donde cada mercado tiene sus leyes y normas. Esta área necesita saber si hubo o no impacto de la inversión en publicidad que se hizo, y para ello necesitan pedir información numérica de otra área.



- b) **Marketing SPI:** son los encargados de recibir la información externa de las variables de mercado, emitida por las empresas de Retail Audit. Además de ser los delegados de reportar los problemas que sobresalen por la información recibida, también son los administradores del sistema BI.
- c) **Marketing Manager:** estos usuarios no pertenecen a un área en particular de marketing, si no que tienen un rango en particular dentro de todo el departamento. Pese a tener un rango superior a los analistas que administran el programa BI, necesitan cierto tipo de restricción sobre todo en la sección de reportes.
- d) **Marketing Finanzas:** los usuarios del área de Finanzas también necesitan información de las otras áreas, ya que un factor muy importante en el movimiento del mercado es la estrategia de la competencia que puede llegar producir una “guerra de precios”.

1.5 Objetivos

Luego de familiarizarse con los procesos del Departamento de Marketing, focalizándose en las tareas efectuadas en la área SPI, es posible definir los objetivos general y específicos que se debe alcanzar con el desarrollo de este trabajo, logrando un beneficio para la compañía.

1.5.1 Objetivo General

Proporcionar un sistema de apoyo a la gestión del área SPI implementando una herramienta BI que redima, en su mayoría, los principales problemas detectados en los procesos desarrollados para la operatividad del área.



De esta forma, se aportarán mejoras en calidad, claridad y precisión tanto en la toma de decisiones como en los procesos del área involucrada.

1.5.2 Objetivos Específicos

Los objetivos específicos del trabajo son los siguientes:

- a) Estudiar y comprender los procesos en la gestión del área SPI, para entender el contexto en el cual se desenvolverá el *software* y así modelar una correcta herramienta BI que permita solucionar los principales problemas.
- b) Modelar y crear una correcta base de datos para reunir toda la información que se utilizará en los procesos del área, y facilitar la búsqueda de la información histórica para los análisis mensuales.
- c) Establecer una interfaz usuaria que se familiarice con las herramientas que utiliza el área, con el fin de favorecer la usabilidad de la solución BI propuesta.
- d) Optimizar los tiempos en la gestión del área, a través de reportes automatizados que permitirán mejorar la exactitud de la información, disminuyendo el error en los cálculos debido a la manipulación excesiva de datos.
- e) Implementar un *test* de validación de la solución propuesta; de esta forma se podrá corroborar que la herramienta otorgada al área de SPI cumple con los requisitos y con la aceptación de los *stakeholders*.



1.5.3 Alcance

El alcance del presente trabajo se puede resumir en lograr una mejora en el desempeño del área SPI. Para ello se podría inferir que mejorando dos de sus principales procesos, el proceso de control histórico de la información y el de reporte, se apreciaría un cambio evidente en la área SPI. A continuación, se explicarán los procesos mencionados con anterioridad.

1.5.3.1 Proceso de control histórico

El proceso de control histórico en este tipo de sistema es fundamental, dado que, al trabajar con información tercerizada, muchas veces ocurre que las empresas externas modifican la información histórica para que los indicadores mensuales le cuadren con lo reportado en el presente mes.

La metodología anterior a la creación del sistema MDB, consistía en crear carpetas nombradas por fecha de recepción de los datos, en donde se guardaba la información y los archivos de otras áreas del respectivo mes, como un archivo Excel extremadamente grande el cual se debía rehacer cada 6 meses debido al tamaño; esto daba problemas a la hora de revisar la historia de los mercados.

1.5.3.2 Proceso de reporte

La gran mayoría de los reportes emitidos por el área P&I se presentan mediante un análisis escrito y con un archivo anexo en formato Excel; el sistema MDB facilita la creación del archivo anexo de manera automática reduciendo drásticamente el tiempo de realizar los cálculos de forma manual.



Para mejorar la ganancia de tiempo en la fabricación de los archivos anexos en los reportes, es necesaria cierta información como saber qué mercados ejecutarán los cálculos y en qué periodo de tiempo. También dependiendo del reporte y del mercado que se seleccione, el tipo de información adicional que se requerirá. Pese a que el reporte que se genera es el mismo para todos los mercados, cada uno tiene diferencias en los mismos productos; por ejemplo la marca Pall Mall en Perú es solo cigarros pero en Chile es cigarros y tabaco, por lo que el reporte puede ser ejecutado para cigarros o tabaco, o en conjunto dependiendo de lo que se solicite. Otro problema encontrado es que un producto en la historia mantiene el nombre, pero puede variar su categoría de Premium a VFM; tomando en cuenta el ejemplo anterior, un producto en un mercado durante el año 2014 era Premium, en el año 2015 bajo de categoría a VFM, y en el año 2016 volvió a ser Premium, por lo cual esta diferenciación historia no puede pasar desapercibida al efectuar un reporte.



Capítulo 2: Marco Teórico

En el presente capítulo se introduce en el concepto de *business intelligence* (BI, inteligencia de negocio) presentando definiciones de los conceptos para luego plantear los diferentes estilos de los *softwares* BI. Continuando con las metodologías para la confección de un *data warehouse*, el modelado de datos multidimensional, ETL, el ciclo de vida del *software*, y finalizar con las heurísticas de Nielsen para el diseño de interfaces usuarias.

2.1 Definición de Business Intelligence

Business Intelligence (BI) es la práctica donde la transformación de los datos duros en información desestructurada se manipula para convertirse en conocimiento, de tal forma que se pueda optimizar el proceso de toma de decisiones en las empresas. A continuación, se citarán definiciones de otros autores:

- a) (*Stackowiak, 2007*) (3) define BI como el proceso de tomar grandes cantidades de datos, analizarlos y presentar un conjunto de informes de alto nivel, que condensa su esencia en la base de las acciones comerciales, lo que permite a la gerencia tomar decisiones de negocios fundamentales.
- b) (*Cui, 2007*) (3) ve BI como una forma y método para mejorar el rendimiento del negocio, al proporcionar poderosas asistencias para que los ejecutivos tomen las decisiones necesarias y puedan tener información procesable a mano.



- c) (Zeng, 2006) (3) definen BI como “el proceso de recopilación, tratamiento y difusión de información que tiene un objetivo, la reducción de la incertidumbre en la toma de todas las decisiones estratégicas”.
- d) (Tvrđíková, 2007) (3) describe que la característica básica de la herramienta de BI es la capacidad de recopilar datos de fuentes heterogéneas, poseer métodos analíticos avanzados y satisfacer las demandas de múltiples usuarios.
- e) (Gartner Group, 1996) (3) presenta el concepto de *BI* como la aplicación de un conjunto de metodologías y tecnologías, tales como J2EE, DOTNET, servicios web, XML, *data warehouse*, OLAP, minería de datos, tecnologías de representación, etc., para mejorar la efectividad de la operación empresarial, y apoyar la gestión / decisión para lograr ventajas competitivas.

En términos generales se puede apreciar de estas definiciones que una aplicación de BI puede lograr un impacto empresarial real, a través de la manipulación de una gran cantidad de datos.

Hoy en día, las empresas han descubierto muchas maneras de cómo utilizar sus activos de información y reportes de operación para así optimizar sus procesos, y finalmente apoyar la toma de decisiones. Es por esto que los proveedores de tecnología BI reaccionan ante la evolución del mercado, construyendo *software* para implementar nuevos tipos de aplicaciones que las empresas requieran. Estos tipos de aplicaciones se convirtieron en productos de *software* centrados en solamente uno de los diferentes estilos de BI que se describen a continuación (4):



- a) Reportería corporativa o empresarial: los reportadores son usados para generar reportes estáticos con un alto grado de control sobre el formato visual, destinados a una amplia distribución entre muchas personas.
- b) Análisis de Cubos: capacidad analítica sobre un subconjunto de datos, dirigida a directivos que requieren un ambiente seguro y sencillo para explorar y analizar sobre un rango delimitado de datos.
- c) Análisis y consultas *Ad Hoc*: las herramientas OLAP relacionales son utilizadas por usuarios avanzados para investigar y analizar toda la base de datos, navegando hasta el nivel más detallado de información, es decir, al nivel de transacción.
- d) Análisis estadístico y minería de datos: aplicación de herramientas matemáticas y estadísticas para encontrar correlaciones, tendencias, proyecciones y análisis financiero. Dirigido a analistas de información avanzados.
- e) Entrega de información y alertas: los motores de distribución de información son utilizados para enviar reportes o alarmas a grandes grupos de usuarios, basándose en suscripciones, itinerarios o acontecimientos.

La mayoría de las empresas líderes han comprado diversas herramientas de BI a distintos proveedores; cada herramienta enfocada a una nueva aplicación BI, y cada una entregando funcionalidad enfocada en un solo estilo de BI. Por lo que no se puede pretender abarcar todos los estilos con una sola herramienta, ya que se vería afectado su desempeño y sin lograr satisfacer las necesidades de los usuarios.

Una forma de organizar los estilos es graficándolos a lo largo de dos dimensiones, donde el eje vertical representa la sofisticación e interactividad, mientras que el eje horizontal muestra el tamaño de la población a quien está dirigido. Esto se refleja en el gráfico 1; las funcionalidades más sofisticadas e interactivas son utilizadas por grupos de usuarios relativamente pequeños, integrados por analistas y usuarios avanzados, cuyo trabajo principal está enfocado en los datos y su análisis. Los estilos menos interactivos proveen datos y los resultados básicos, aplicables a poblaciones grandes de usuarios, que van desde directores generales hasta el grueso del personal operativo.

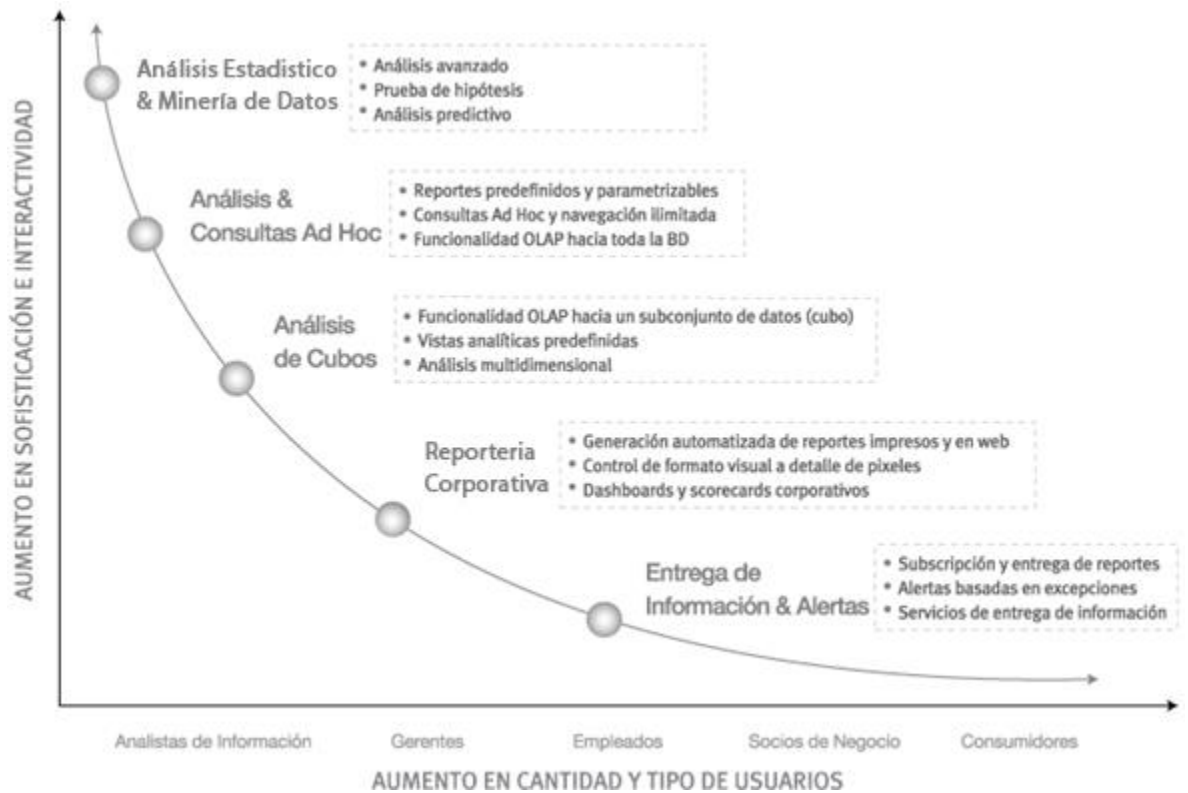


Gráfico 1: Funcionalidad vs. Stakeholders (4)



2.2 Herramientas BI

En la actualidad existe una infinidad de *software* enfocados en BI; por una parte, están los llamados *open source*, como por ejemplo *JasperReports*, *LogiReport*, *Pentaho*, o *Eclipse BIRT Project*. Por otra parte, se encuentran los productos comerciales como ApeSoft, Microsoft Excel, Microsoft SQL Server, MicroStrategy o SAP HANA Enterprise (5).

Cada *software* BI intenta focalizarse en uno de los estilos o propiedades que destacan las herramientas de inteligencia de negocios, como el cuadro de mando, *Digital Dashboard*, OLAP, reportería o minería de datos. Claramente es muy complejo intentar abarcar todas las posibilidades de la herramienta BI en un solo *software*; es por esto que varias empresas utilizan más de una para la toma de decisiones.

2.3 Metodologías para un Data Warehouse

En esta sección se presentan dos metodologías para la confección de un *data warehouse* (DW) para el desarrollo de una herramienta BI: la de Bill Inmon (esquema *top-down*) y la de Ralph Kimball (*Bottom-Up*).

En la primera se ve la necesidad de transferir la información desde los diferentes sistemas de las organizaciones a un lugar centralizado donde los datos puedan ser utilizados para el análisis. Insiste además en que este ha de tener las siguientes características (6):



- a) Orientado a temas: los datos en la base de datos están organizados de manera que todos los elementos relativos al mismo evento u objeto del mundo real queden unidos entre sí.
- b) Integrado: la base de datos contiene los datos de todos los sistemas operacionales de la organización, y dichos datos deben ser consistentes.
- c) No volátil: la información no se modifica ni se elimina, una vez almacenado un dato, éste se convierte en información de sólo lectura, y se mantiene para futuras consultas.
- d) Variante en el tiempo: los cambios producidos en los datos a lo largo del tiempo quedan registrados para que los informes que se puedan generar reflejen esas variaciones.

Los datos son extraídos de los sistemas operacionales y cargados en las áreas de *stage*, donde son validados en el DW y consolidados en los *data marts* (subconjunto de datos de un DW).

En cuanto a la segunda metodología, esta se basa en el denominado Ciclo de Vida Dimensional del Negocio, fundamentado en cuatro principios básicos (7):

- a) Centrarse en el negocio: hay que concentrarse en la identificación de los requerimientos del negocio y su valor asociado, y usar estos esfuerzos para desarrollar relaciones sólidas con el negocio, agudizando el análisis del mismo y la competencia consultiva de los implementadores.
- b) Construir una infraestructura de información adecuada: diseñar una base de información única, integrada, fácil de usar, de alto rendimiento donde se



reflejará la amplia gama de requerimientos de negocio identificados en la empresa.

- c) Realizar entregas en incrementos significativos: crear el DW en incrementos entregables en plazos de 6 a 12 meses. Cada elemento posee un valor para el negocio, que facilita la determinación del orden de aplicación de los incrementos. En esto, la metodología se parece a las metodologías ágiles de construcción de *software*.
- d) Ofrecer la solución completa: tener todos los elementos necesarios para entregar valor a los usuarios de negocios. Esto significa tener un DW sólido, bien diseñado, con calidad probada, y accesible. También se deberá entregar herramientas de consulta, aplicaciones para informes y análisis avanzado, capacitación, soporte, sitio web y documentación.

Este enfoque sigue un esquema de trabajo de tipo *bottom-up*, pues al final el DW no es más que la unión de los diferentes *data marts*, que están estructurados de una forma común. Esta característica le hace más flexible y sencillo de implementar. En la figura 1 se muestra una comparación simplificada de las dos metodologías que se han planteado sobre la confección de un DW.



Figura 2: Diagrama comparativo entre las metodologías de Inmon vs. Kimball (7)

2.4 Modelado de datos multidimensional

El modelamiento dimensional es una técnica para modelar bases de datos de apoyo a la gestión. Una base de datos multidimensional, es aquella que almacena sus datos con varias dimensiones, es decir, en vez de un valor, se encuentran varios dependiendo de los "ejes" definidos, o dicho de otra forma una base de datos de estructura basada en dimensiones.

Cuando una base de datos puede ser visualizada como un cubo de tres o más dimensiones, es más fácil para el usuario organizar la información e imaginarse en ella "cortando" y "rebanando" el cubo a través de cada una de sus dimensiones, para buscar la información deseada.

Las bases de datos multidimensionales proveen una estructura que permite tener acceso flexible a los datos, para explorar y analizar sus relaciones; éstas se pueden visualizar como un cubo multidimensional, en donde las variables asociadas existen a lo

largo de varios ejes o dimensiones, y la intersección de las mismas representa la medida, indicador o hecho que se está evaluando.

Las bases de datos multidimensionales tienen tres variantes posibles de esquemas que permiten realizar consultas de soporte a la decisión (8):

- a) Estrella (*Star Scheme*): es un modelo de datos que tiene una tabla central que contiene los datos para el análisis, rodeada de las tablas de dimensiones. Es decir, consiste en una tabla de hechos en el centro para el análisis y una o varias tablas de dimensión que participan de la descripción de ese hecho. La tabla de hechos es la única tabla que tiene relaciones que la conectan con otras tablas. El resto de tablas del esquema únicamente hacen *relacion* con esta tabla de hechos. En la figura 2 se muestra un esquema de estrella con una sola tabla de hechos y cuatro tablas de dimensiones.

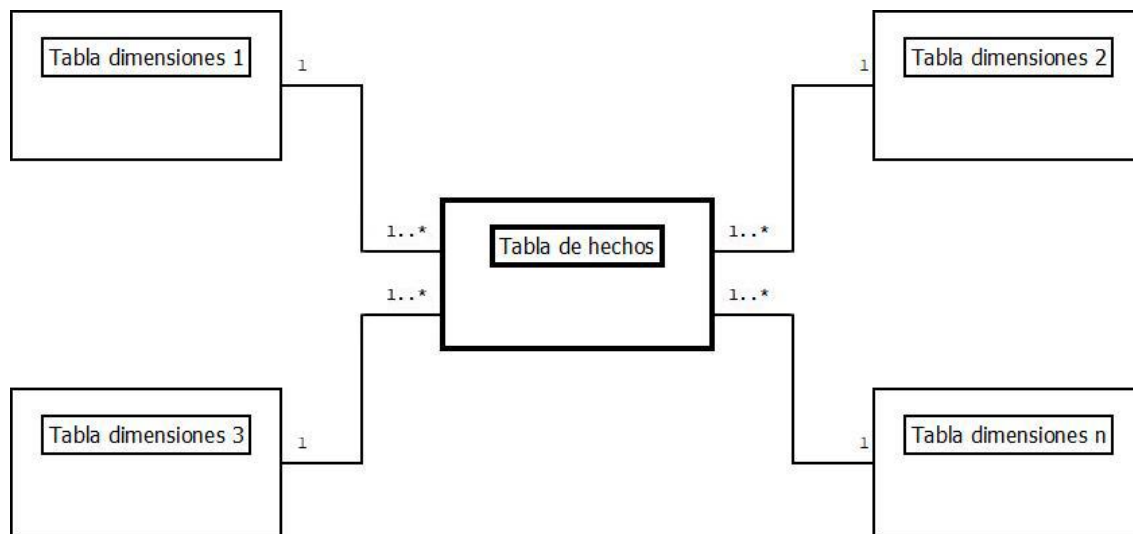


Figura 3: Ejemplo de Modelo de datos Estrella (9)

b) Copo de nieve (*Snowflake Scheme*): es una estructura algo más compleja que el esquema en estrella. Se da cuando todas las dimensiones de la tabla de hechos tienen varios niveles para su descripción. La finalidad es normalizar las tablas y así reducir el espacio de almacenamiento al eliminar la redundancia de datos; pero tiene la contraparte de generar bajos rendimientos al tener que crear más tablas dentro de las dimensiones y ejecutar más *joins* entre las tablas al procesar, por lo que tiene un impacto directo sobre el rendimiento. En la figura 3 se puede observar un ejemplo sencillo del modelo copo de nieve.

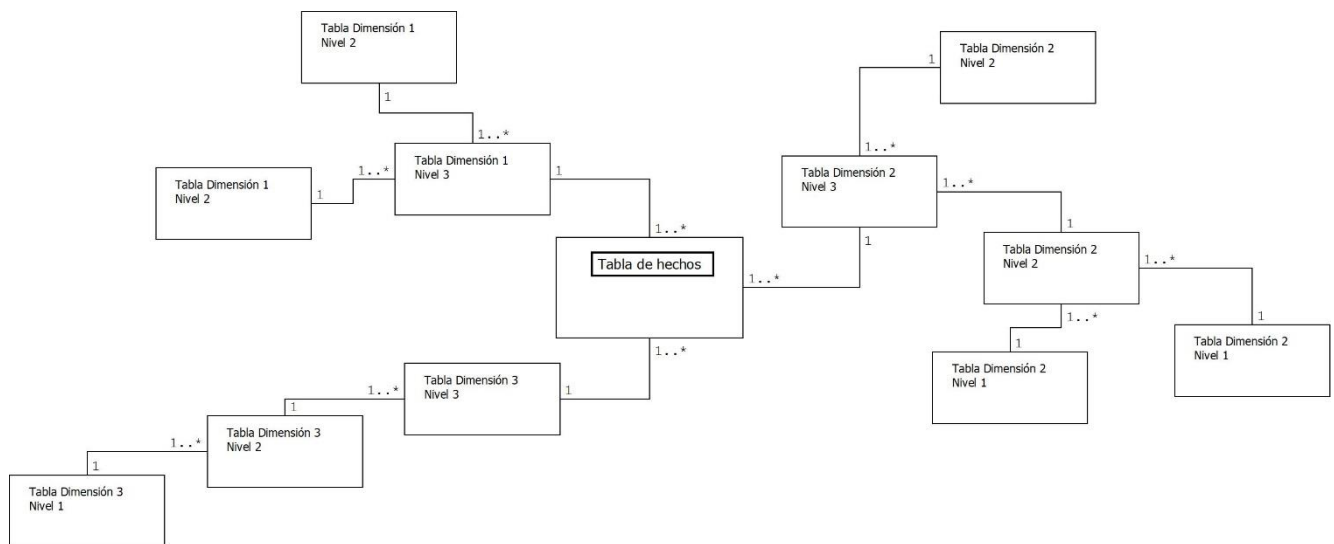


Figura 4: Ejemplo de Modelo Copo de Nieve (9)

c) Constelación o copo de estrellas (*Starlike Scheme*): es más complejo que las otras arquitecturas debido al hecho de que contiene múltiples tablas de hechos. Con esta solución las tablas de dimensiones pueden estar compartidas entre más que una tabla de los hechos, dando flexibilidad al modelado. Sin embargo, el problema es que cuando el número de las tablas vinculadas

aumenta, la arquitectura puede llegar a ser muy compleja y difícil para mantener. En la figura 4 se puede observar un ejemplo sencillo del modelo copo de estrellas.

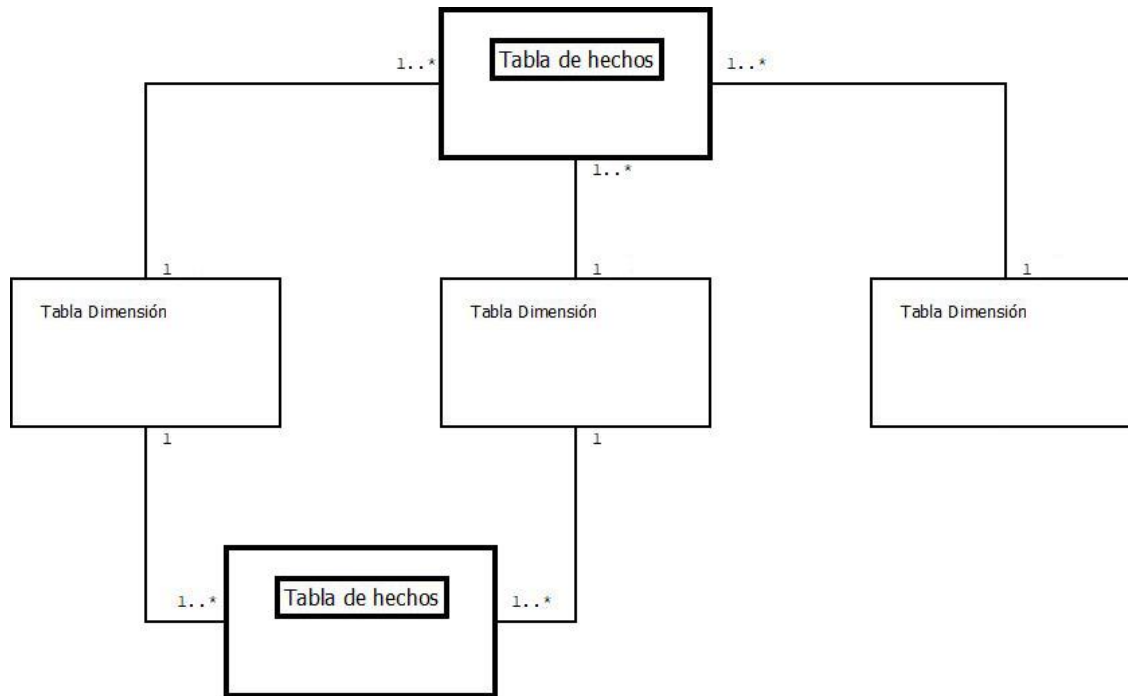


Figura 5: Ejemplo de Modelo Constelación (9)

Los esquemas ya mencionados pueden ser implementados de diversas maneras; independiente al tipo de arquitectura, requieren que toda la estructura de datos esté desnormalizada o semidesnormalizada, para evitar *joins* complejos para acceder a la información, con el fin de agilizar la ejecución de consultas.

2.5 OLAP

La gran mayoría de los *software* BI son del tipo OLAP (*On-Line Analytical Processing*), debido a la rápida respuesta frente una consulta. Para este tipo de análisis



se utilizan estructuras de base de datos multidimensionales, necesarias por la superioridad en velocidad de respuesta respecto a las bases relacionales (10).

La expresión OLAP apareció en 1993, cuando E.F. Codd la describió con doce reglas de evaluación (11):

- a) Vista conceptual multidimensional
- b) Transparencia
- c) Accesibilidad
- d) Rendimiento consistente de los informes
- e) Arquitectura cliente-servidor
- f) Dimensionalidad genérica
- g) Manejo dinámico de matriz dispersa
- h) Soporte multiusuario
- i) Operaciones cruzadas dimensionales sin restricciones
- j) Manipulación de datos intuitiva
- k) Informes flexibles
- l) Dimensiones ilimitadas y niveles de agregación

Lo anterior es una enumeración algo larga, y está más relacionada a una evaluación de *software* más que una descripción de lo que es OLAP. Sin embargo, Nigel Pendse presenta una definición muy breve de lo que es OLAP: “Análisis rápido de información multidimensional compartida” (12).

Existen diversos tipos de sistemas OLAP, que en su mayoría se clasifican en MOLAP, ROLAP y HOLAP, los cuales se explican a continuación:

- a) MOLAP: esta arquitectura usa bases de datos multidimensionales para apoyar el análisis; su principal premisa es que el OLAP está mejor implantado

almacenando los datos de dicha forma. El sistema MOLAP utiliza una arquitectura de dos niveles: la base de datos multidimensionales y el motor analítico (13). La base de datos multidimensional es la encargada del manejo, acceso y obtención del dato (ver figura 5).

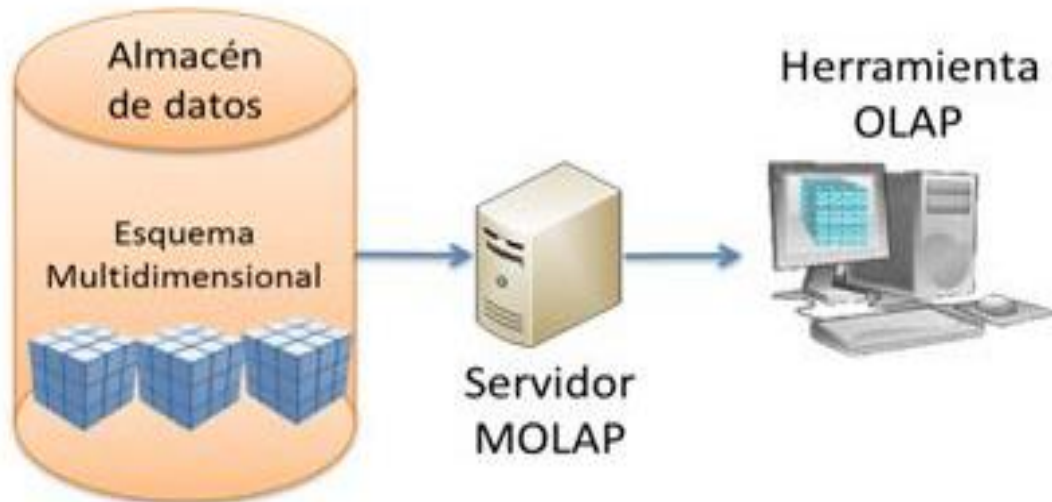


Figura 6: Ejemplo de arquitectura MOLAP (14)

Tras poblar esta estructura multidimensional, se generan índices y algoritmos de tablas para mejorar los tiempos de accesos a las consultas. La arquitectura MOLAP requiere cálculos intensivos de compilación; lee de datos precompilados y tiene capacidades limitadas de crear agregaciones dinámicamente o de hallar *ratios* que no se hayan precalculados y almacenados previamente.

b) ROLAP: esta arquitectura cree que las capacidades OLAP están perfectamente implantadas sobre bases de datos relacionales. El sistema ROLAP utiliza una arquitectura de tres niveles. El nivel de almacenamiento maneja los requerimientos de la base de datos, y el motor ROLAP apoya la

funcionalidad analítica (15), usando tablas para el manejo, acceso y obtención del dato. El nivel de aplicación es el motor que ejecuta las consultas multidimensionales de los usuarios (ver figura 6). El motor ROLAP se integra con niveles de presentación, a través de los cuales los usuarios realizan los análisis OLAP, que transforma dinámicamente sus consultas a *queries* SQL. Se ejecutan estas *queries* SQL en las bases de datos relacionales, y sus resultados se relacionan mediante tablas cruzadas y conjuntos multidimensionales para devolver los resultados a los usuarios.

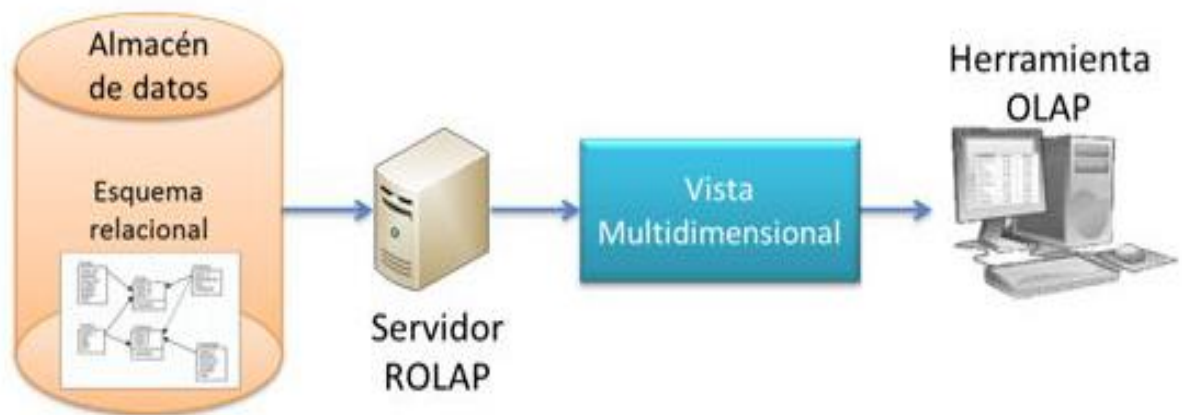


Figura 7: Ejemplo de arquitectura ROLAP (14)

La arquitectura ROLAP es capaz de usar datos precalculados si estos están disponibles, o de generar dinámicamente los resultados desde los datos elementales si es preciso. Accede directamente a los datos y soporta técnicas de optimización de accesos para acelerar las consultas. Estas optimizaciones son, entre otras, particionamiento de los datos a nivel de aplicación, soporte a la

desnormalización y *joins* múltiples. En la tabla 1 se presenta un cuadro comparativo entre ROLAP y MOLAP.

Criterio de comparación	MOLAP	ROLAP
Rendimiento	Ofrece un rendimiento teóricamente superior al reorganizar físicamente los datos	El rendimiento es peor, y si no se organizan los datos un poco, es desastroso
Tamaño	Al ser una estructura propietaria, se suelen aplicar técnicas de compresión. Pero cuidado, una mala planificación de las dimensiones de los cubos y se puede gastar muchísimo espacio.	No se crean nuevas estructuras, aunque si no se han hecho antes deberían montar un esquema estrella normalizado, lo que consumirá espacio
Funcionalidad	Normalmente dan más funcionalidad, permitiendo operaciones específicas sobre el cubo para potenciar su análisis	Se basa en SQL, aunque en las últimas versiones de algunas bases de datos ya poseen instrucciones específicas como CUBE o ROLLUP que imitan la funcionalidad de una herramienta OLAP
Escalabilidad	Aquí flojea. Cada cubo ha de crearse bajo demanda, y refrescarse, etc.	En teoría es más escalable y maneja mejor grandes volúmenes de datos con dimensiones de alta cardinalidad.

Tabla 1: MOLAP vs. ROLAP (16)

c) HOLAP: es una combinación de los dos tipos de sistema anteriores (ver figura 7). Los datos agregados y precalculados se almacenan en estructuras multidimensionales (MOLAP) y los de menor nivel de detalle en el relacional (ROLAP). Requiere un buen trabajo de análisis para identificar cada tipo de dato.

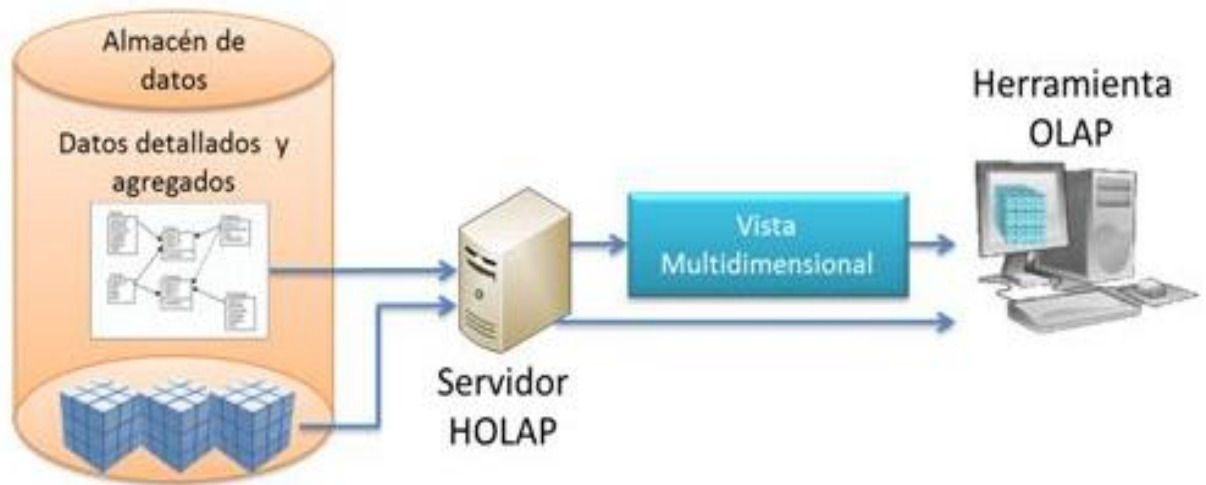


Figura 8: Ejemplo de arquitectura HOLAP (14)

2.6 ETL

ETL son las siglas en inglés de Extraer, Transformar y Cargar (*Extract, Transform and Load*). Es el proceso que permite a las organizaciones mover datos desde múltiples fuentes, reformatear y limpiarlos, y cargarlos en otra base de datos, *data mart* o *data warehouse* para analizar, o en otro sistema operacional para apoyar un proceso de negocio (17). A continuación, se explica brevemente cada una de las etapas de ETL.

La primera parte del proceso ETL consiste en extraer los datos desde los sistemas de origen. La mayoría de los proyectos de almacenamiento de datos fusionan datos



provenientes de diferentes sistemas de origen. Cada sistema separado puede usar una organización diferente de los datos o formatos distintos. Los formatos de las fuentes normalmente se encuentran en bases de datos relacionales o archivos planos, pero pueden incluir bases de datos no relacionales u otras estructuras diferentes. La extracción convierte los datos a un formato preparado para iniciar el proceso de transformación (18).

Una parte intrínseca del proceso de extracción es la de analizar los datos extraídos, de lo que resulta un chequeo que verifica si los datos cumplen la pauta o estructura que se esperaba. De no ser así los datos son rechazados.

La segunda parte sería la fase de transformación de un proceso de ETL donde aplica una serie de reglas de negocio o funciones sobre los datos extraídos para convertirlos en datos que serán cargados. Algunas fuentes de datos requerirán alguna pequeña manipulación de los datos; no obstante, en otros casos pueden ser necesarias aplicar algunas de las siguientes transformaciones (18):

- Seleccionar sólo ciertas columnas para su carga (por ejemplo, que las columnas con valores nulos no se carguen).
- Traducir códigos (por ejemplo, si la fuente almacena una “H” para Hombre y “M” para Mujer pero el destino tiene que guardar “1” para Hombre y “2” para Mujer).
- Codificar valores libres (por ejemplo, convertir “Hombre” en “H” o “Sr” en “1”).
- Obtener campos derivados (por ejemplo, $total_venta = cantidad * precio$).
- Unir datos de múltiples fuentes (por ejemplo, búsquedas, combinaciones, etc.).



- Calcular totales de múltiples filas de datos (por ejemplo, ventas totales de cada región).
- Generar campos clave en el destino.
- Transponer o pivotear (girando múltiples columnas en filas o viceversa).
- Dividir una columna en varias (por ejemplo, columna “Nombre: García, Miguel”; pasar a dos columnas “Nombre: Miguel” y “Apellido: García”).
- Aplicar, de forma simple o compleja, una validación de datos, y la consiguiente aplicación de la acción que en cada caso se requiera:
 - Datos correctos: entregar datos a la siguiente etapa (Carga).
 - Datos erróneos: ejecutar políticas de tratamiento de excepciones (por ejemplo, rechazar el registro completo, dar al campo erróneo un valor nulo o un valor centinela).

Por último en la tercera parte, la fase de carga, es el momento en el cual los datos de la fase anterior (transformación) son cargados en el sistema de destino. Dependiendo de los requerimientos de la organización, esta fase puede abarcar una amplia variedad de acciones diferentes. En algunas bases de datos se sobrescribe la información antigua con nuevos datos. Los DW deben mantener un historial de los registros de manera que se pueda hacer una auditoría de los mismos, y disponer de un rastro de toda la historia de un valor a lo largo del tiempo (18).

Existen dos formas básicas de desarrollar el proceso de carga:

- Acumulación simple: es la más sencilla y común, y consiste en realizar un resumen de todas las transacciones comprendidas en el período de tiempo



seleccionado y transportando el resultado como una única transacción hacia el DW, almacenando un valor calculado que consistirá típicamente en un resumen o promedio de la magnitud considerada.

- *Rolling*: se aplica en los casos en que se opta por mantener varios niveles de granularidad. Para ello se almacena información resumida a distintos niveles, correspondientes a distintas agrupaciones de la unidad de tiempo o diferentes niveles jerárquicos en alguna o varias de las dimensiones de la magnitud almacenada (por ejemplo, totales diarios, totales semanales, totales mensuales, etc.).

La fase de carga interactúa directamente con la base de datos de destino. Al realizar esta operación se aplicarán todas las restricciones y *triggers* (disparadores) que se hayan definido en ésta, que de estar bien definidos contribuyen a que se garantice la calidad de los datos en el proceso ETL; y deben ser tomados en cuenta al momento de efectuar la carga de los datos.

2.7 Modelo de ciclo de vida

Un modelo de ciclo de vida describe las fases de un proyecto y su orden de ejecución, desde la concepción inicial, diseño, desarrollo, puesta en marcha y posterior mantenimiento. Existen varios modelos de ciclo de vida, entre los cuales están los siguientes: modelo en cascada, modelo en V, modelo iterativo, modelo incremental, modelo en espiral y modelo de prototipos (19).

El modelo en cascada, representado en la figura 8, fue propuesto por Winston Royce en el año 1970. Es un ciclo de vida que admite iteraciones, después de cada etapa se realiza una o varias revisiones para comprobar si se puede pasar a la siguiente. Es un

modelo rígido, poco flexible, y con muchas restricciones, aunque fue uno de los primeros y sirvió de base para el resto de los modelos de ciclo de vida. Una de sus ventajas, además de su planificación sencilla, es la de proveer un producto con un elevado grado de calidad sin necesidad de un personal altamente calificada. Se puede considerar como inconvenientes: la necesidad de contar con todos los requerimientos (o la mayoría) al comienzo del proyecto, y si se han cometido errores y no se detectan en la etapa inmediata siguiente, es costoso y difícil volver atrás para realizar la corrección posterior.

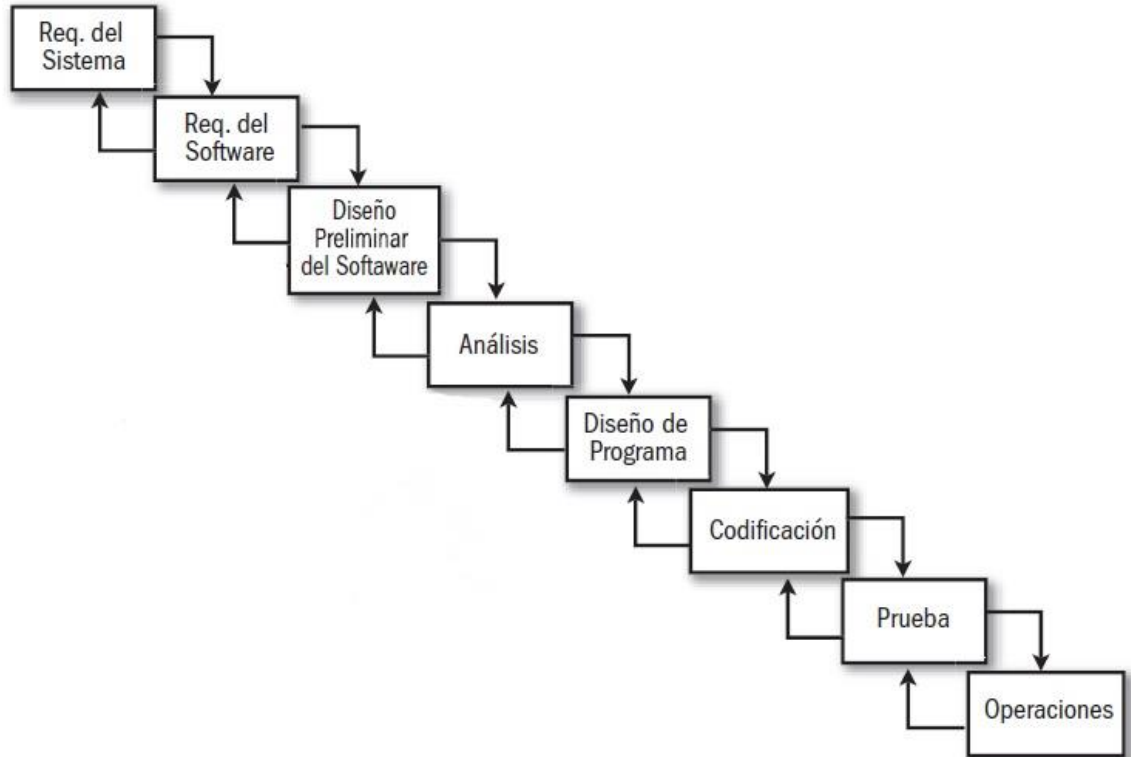


Figura 9: Modelo del ciclo de vida en cascada (20)

El modelo del ciclo de vida en V representado en la figura 9, fue diseñado por Alan Davis, y contiene las mismas etapas que el ciclo de vida en cascada. A diferencia de este, el modelo en V se le agregaron dos subetapas de retroalimentación entre las etapas

de análisis y mantenimiento, y entre las de diseño y *debugging*. Las ventajas y desventajas de este modelo son las mismas del ciclo de vida en cascada, con el agregado de los controles cruzados entre etapas para lograr una mayor corrección de errores.

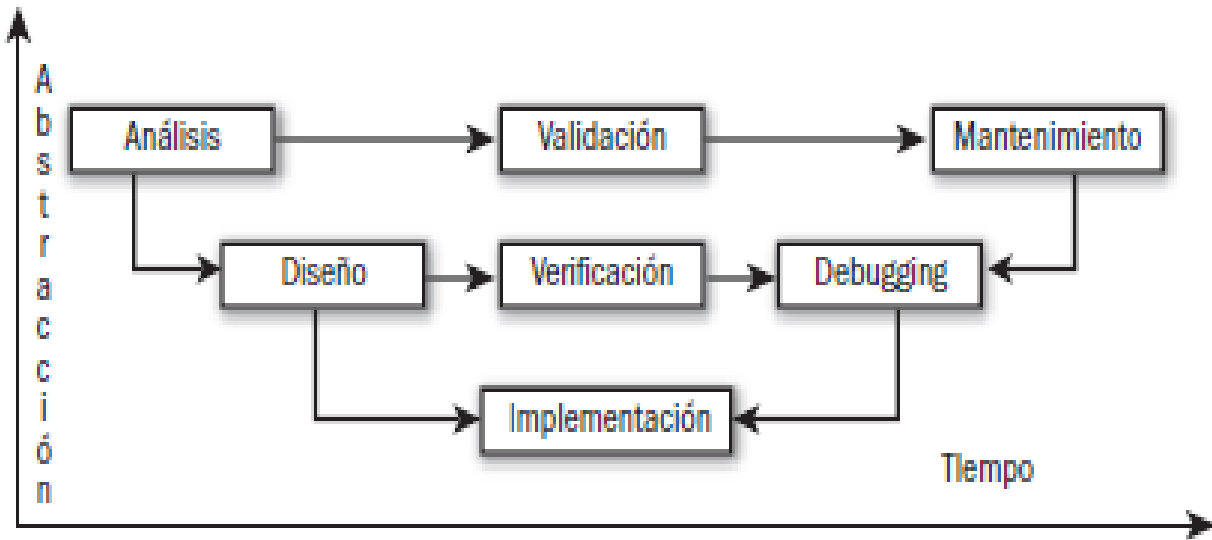


Figura 10: Modelo del ciclo de vida en V (20)

El modelo ciclo de vida iterativo representado en la figura 10, también derivado del ciclo de vida en cascada, busca reducir el riesgo que surge entre las necesidades del usuario y el producto final por malos entendidos durante la etapa de solicitud de requerimientos. Es la iteración de varios ciclos de vida en cascada que al final de cada iteración se le entrega al cliente una versión mejorada o con mayores funcionalidades del producto. El cliente es quien luego de cada iteración, evalúa el producto y lo corrige o propone mejoras. Estas iteraciones se repiten hasta obtener un producto que satisfaga al cliente.

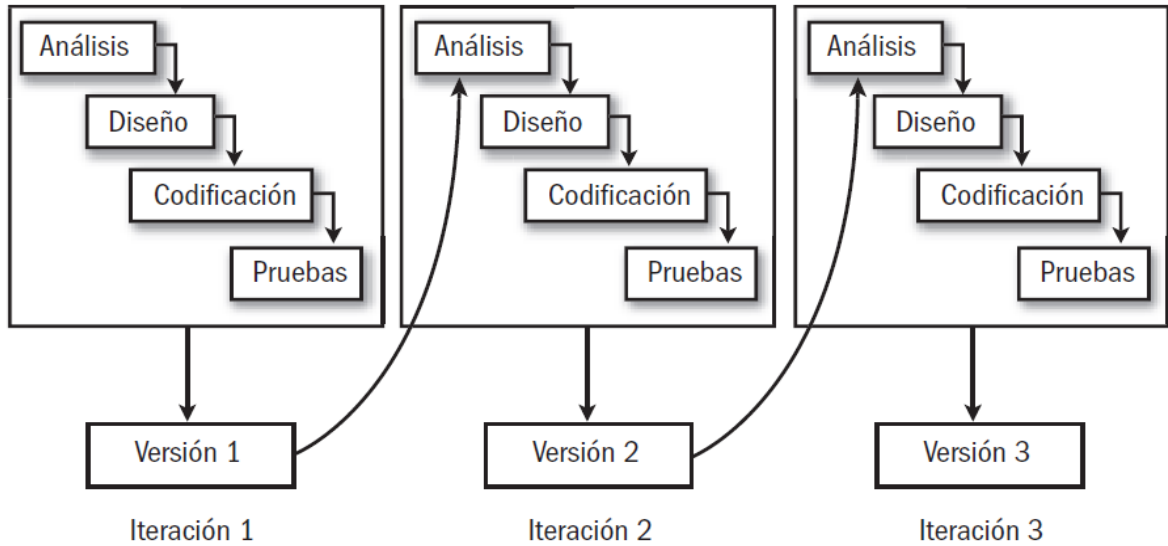


Figura 11: Modelo del ciclo de vida iterativo (20)

El modelo incremental representado en la figura 11, se basa en la filosofía de construir incrementando las funcionalidades del programa. Se realiza construyendo por módulos que cumplen las diferentes funciones del sistema. Esto permite ir aumentando gradualmente las capacidades del software. También es derivado del ciclo de vida en cascada, pero aplicándose este ciclo en cada funcionalidad del programa a construir. Al final de cada ciclo se entrega una versión al cliente que contiene una nueva funcionalidad. Este ciclo de vida permite realizar una entrega al cliente antes de terminar por completo el proyecto.

Para el presente trabajo se utilizó el modelo de ciclo de vida incremental que sirvió para mantener un desarrollo claro y sin confusiones en los requerimientos de los *stakeholders*.

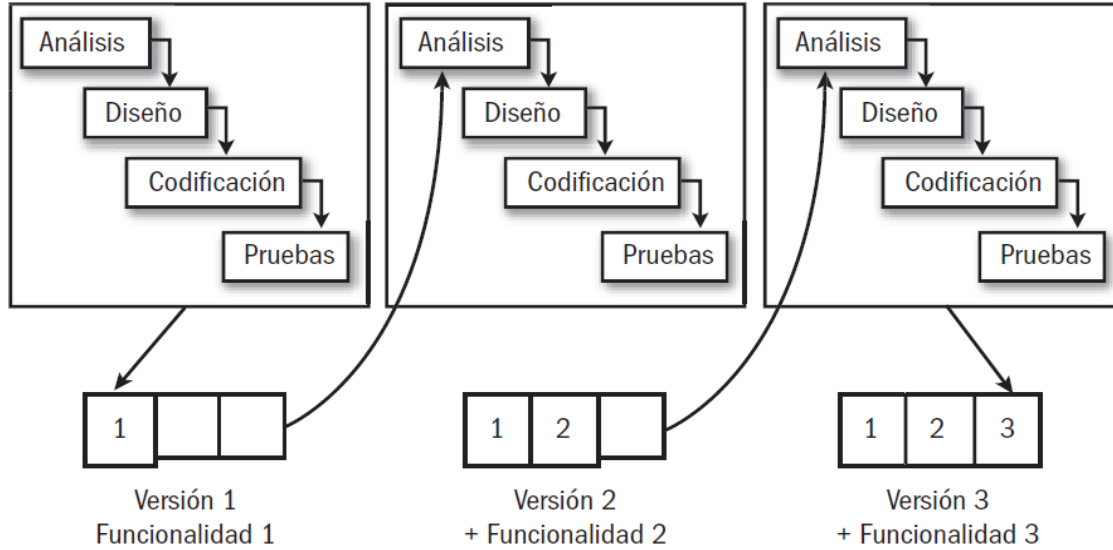


Figura 12: Modelo del ciclo de vida incremental (20)

Las fases utilizadas en el proyecto fueron:

- Carga de la información mensual: las actividades en esta fase son aquellas que se encuentran relacionadas con la información recibida de fuentes externas e internas mensualmente que requieren ser cargadas al sistema.
- Control de la información: son todas las tareas relacionadas a la comprobación de la información recibida, de tal manera que cuadren los datos históricos registrados y que la información cargada coincida con la información de origen.
- Análisis de información: abarca todas las tareas que manipulan la información de tal manera que proporcionen análisis o reportes gerenciales.
- Adicionales: esta etapa considera todas las actividades que luego del análisis requieran modificaciones de la información de origen y reprocesar nuevamente todo el ciclo.



2.8 Heurísticas de Nielsen

Hoy en día existen muchos tipos de *tests* y pruebas para evaluar la usabilidad y asegurar una buena experiencia del usuario al enfrentarse a una interfaz. En las interacciones humano-computador, se siguen varios pasos para crear sistemas que sean prácticos para el usuario a nivel de usabilidad. En el paso de evaluación, se realizan dos tipos de pruebas, de usabilidad y de expertos. En esta última se utilizan las heurísticas creadas por Jakob Nielsen en 1995 para evaluar el diseño de la interfaz de usuario. Estas tratan de diez sencillas normas que toda interfaz interactiva debería cumplir, para garantizar que no se incurrirá en problemas de usabilidad a los usuarios.

A continuación, se enumerarán las 10 heurísticas de Jakob Nielsen (21):

- a) Visibilidad del estado del sistema: el sistema siempre debería mantener informados a los usuarios de lo que está ocurriendo, a través de retroalimentación apropiada dentro de un tiempo razonable.
- b) Relación entre el sistema y el mundo real: el sistema debería hablar el lenguaje de los usuarios mediante palabras, frases y conceptos que sean familiares al usuario, más que con términos relacionados con el sistema. Seguir las convenciones del mundo real, haciendo que la información aparezca en un orden natural y lógico.
- c) Control y libertad del usuario: hay ocasiones en que los usuarios elegirán las funciones del sistema por error y necesitarán una “salida de emergencia” claramente marcada para dejar el estado no deseado al que accedieron, sin tener que pasar por una serie de pasos. Se deben apoyar las funciones de deshacer y rehacer.



- d) Consistencia y estándares: los usuarios no deberían cuestionarse si acciones, situaciones o palabras diferentes significan en realidad la misma cosa; siga las convenciones establecidas.
- e) Prevención de errores: mucho mejor que un buen diseño de mensajes de error es realizar un diseño cuidadoso que prevenga la ocurrencia de problemas.
- f) Reconocimiento antes que recuerdo: se deben hacer visibles los objetos, acciones y opciones, El usuario no tendría que recordar la información que se le da en una parte del proceso, para seguir adelante. Las instrucciones para el uso del sistema deben estar a la vista o ser fácilmente recuperables cuando sea necesario.
- g) Flexibilidad y eficiencia de uso: la presencia de aceleradores, que no son vistos por los usuarios novatos, puede ofrecer una interacción más rápida a los usuarios expertos que la que el sistema puede proveer a los usuarios de todo tipo. Se debe permitir que los usuarios adapten el sistema para usos frecuentes.
- h) Estética y diseño minimalista: los diálogos no deben contener información que es irrelevante o poco usada. Cada unidad extra de información en un diálogo, compite con las unidades de información relevante y disminuye su visibilidad relativa.
- i) Ayudar a los usuarios a reconocer, diagnosticar y recuperarse de errores: los mensajes de error se deben entregar en un lenguaje claro y simple, indicando en forma precisa el problema y sugerir una solución constructiva al problema.
- j) Ayuda y documentación: incluso en los casos en que el sistema pueda ser usado sin documentación, podría ser necesario ofrecer ayuda y



documentación. Dicha información debería ser fácil de buscar, estar enfocada en las tareas del usuario, con una lista concreta de pasos a desarrollar y no ser demasiado extensa.



Capítulo 3: Solución Propuesta

La solución propuesta a los problemas presentados anteriormente, es la creación de una herramienta BI que apoye el control y seguimiento de la información base, con la cual trabaja la área de marketing SPI.

3.1 Descripción de la herramienta BI

La herramienta llamada MDB (*Marketing Data Base*), propuesta como solución, está basada en VBA (Visual Basic for Applications) comúnmente utilizado en los programas de Microsoft Office. Considerando que el Departamento de Marketing se encuentra muy familiarizado con la utilización de Excel, debido a la versatilidad que posee este programa para la manipulación de datos en plantillas de cálculos, es que se usará en Excel como su plataforma base. El proyecto MDB fue dividido en dos archivos Excel, uno de visualización y consulta llamado “MDB Information”, y otro de manipulación de la información denominado “MDB Administration”; estos archivos son complementarios en el sistema.

Se optó por separar en dos el proyecto MDB principalmente por el peso que poseía el archivo, y dado que la empresa tiene una política de HomeOffice, que permite la utilización del acceso remoto a los servidores. Esta política propone a los usuarios el trabajar una vez al mes desde un enlace externo a la compañía, por lo que la calidad de la conectividad impacta fuertemente la actualización del programa; este argumento surgió poniendo en práctica la herramienta. Asimismo, otros departamentos de la área de marketing también consultan MDB utilizando una conexión remota de seguridad con



“Junos Pluse” (*software* que permite acceso remoto y una conectividad segura a los servidores de la empresa) la cual es extremadamente lenta, por tanto el tamaño del archivo es crucial. Estas dos partes se describen brevemente a continuación:

- a) El MDB Administration es utilizado por el equipo de P&I (Planning & information), el cual recibe toda la información que es cargada al sistema. Es responsable de la veracidad de esta, corroborando que los datos presentados como avances sean correctos.
- b) El MDB Information es utilizado por gran parte del Departamento de marketing, logrando abarcar áreas que se encuentran en Argentina, Bolivia, Chile, Perú, Paraguay, y Uruguay. Además de entregar información de los mercados, también generará reportes personalizados los cuales serán usados mensualmente por el equipo P&I. Estos reportes han logrado optimizar los procesos, reduciendo errores y el tiempo en el procesamiento de los datos.

3.2 Elección de la metodología de trabajo

En esta sección, se planteará la metodología la cual se llevará a cabo para el desarrollo de la solución propuesta en el presente trabajo.

La herramienta propuesta se focalizará en dos tipos de sistemas: Reportería corporativa o empresarial, y Análisis estadístico y minería de datos, los que ya fueron explicados previamente en la sección 2.1; esto, debido a los requisitos impuestos por el área de SPI. Se utilizará la metodología de Kimball donde hay que centrarse en el negocio identificando los requerimientos y su valor asociado; construir una



infraestructura de información adecuada en donde el diseño de la base de datos debe estar integrada y ser de fácil uso; realizar entregas incrementales en la creación del DW; y por último ofrecer una solución completa que entregue los elementos necesarios para otorgar valor al negocio.

El sistema que se utilizó es de una arquitectura HOLAP, de tal manera que se almacenen los datos principales en una estructura multidimensional, y los de menor nivel de detalle, de manera relacional.

Todo este sistema se vio acompañado de una interfaz gráfica la cual se desarrolló en base a las heurísticas de Nielsen vistas en la sección 2.8.

3.3 Diseño del modelo de datos

Antes sumergirse en el diseño de la base de datos, se recopiló la información necesaria para poder visualizar los requisitos del sistema, la cual se maneja en el área de SPI:

- a) Mercados o países.
- b) Descripción de cada SKU (Producto)
- c) Volúmenes (diversas fuentes), actuales o reales y proyecciones futuras, que se disgrega por SKU y mercado.
- d) Distribuciones e indicadores métricos
- e) Precios entregados por el Departamento de Finanzas.

La relación entre dichos campos o datos es teórica, ya que todas las fuentes reportan valores de los mismos productos. Pero la dificultad es que cada fuente maneja sus

propias nomenclaturas para identificar los SKU. En la figura 12 se presenta el modelo de la base de datos del tipo constelación, obtenido al analizar los datos recopilados de las diversas fuentes de información que se envían al área SPI.

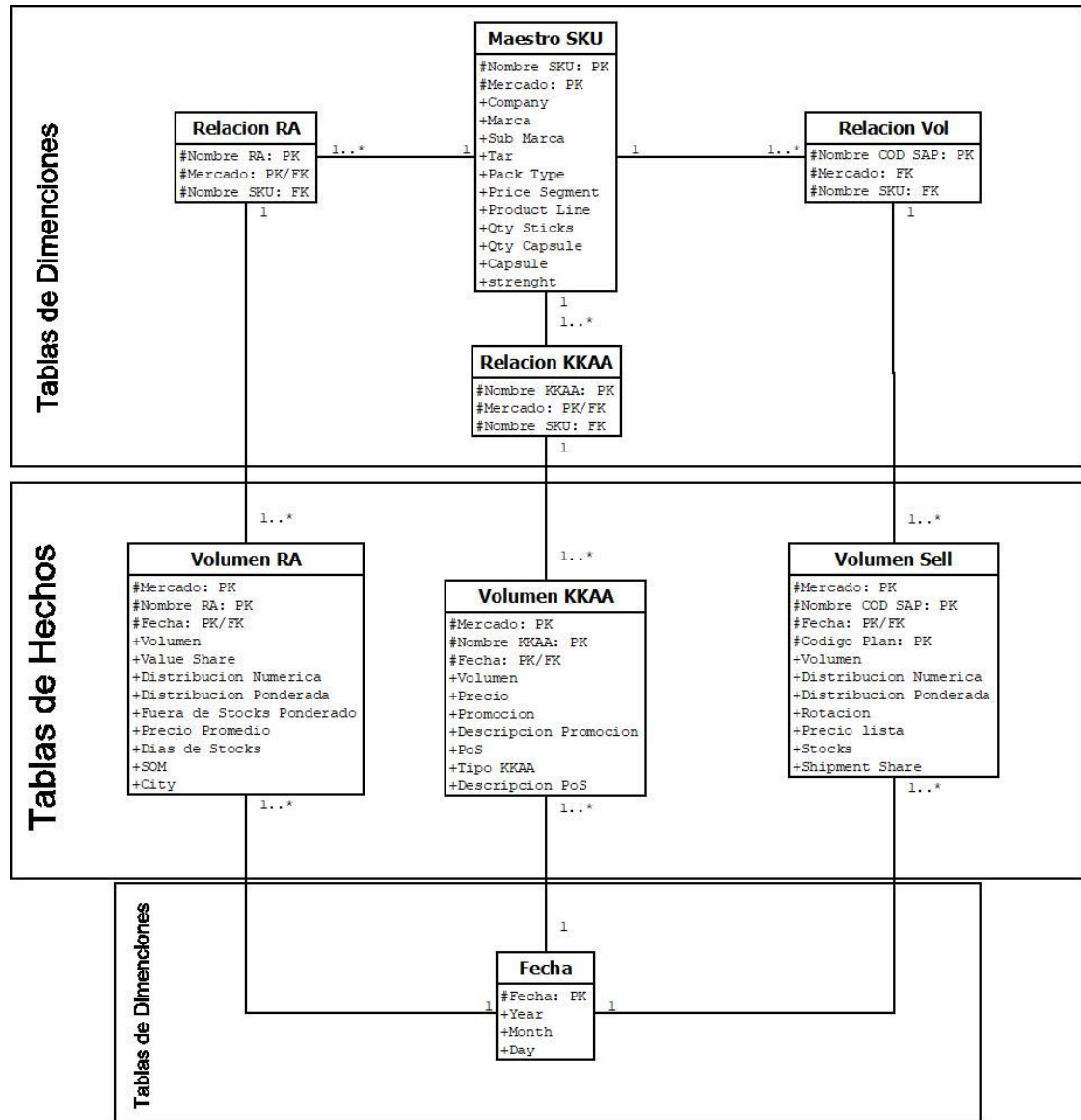


Figura 13: Diagrama de la Base de Datos

Para comprender mejor por que se generó este modelo de datos, se debe entender cómo es que recibe la documentación de origen el área SPI. La información de la tabla



de hechos “Volumen *Sell*” proviene de un sistema interno de la compañía el cual lleva un control de las ventas *Sell In*; este sistema se complementa con las ventas *Sell Out* de cada mercado, las cuales son entregadas por cada distribuidor. Respecto a los datos de la tabla de hechos “Volumen *KKAA*”, provienen de cada mercado, la cual se reporta con distintos nombres de productos dependiendo del mercado y de cada cadena de bencineras que reporte la información, por lo que se tiene un mismo producto, pero con tantos nombres identificadores como compañías de petroleras de cada país. Por último, la tabla de hechos “Volumen *RA*” obtiene su información de cada compañía de *Retail Audit* contratada en cada país, y aunque en algunos países sea la misma compañía, los identificadores de los productos son asignados localmente, por lo que los nombres de los productos por país son distintos, pero pueden tener la misma descripción. Por todo lo anterior, el modelo de datos que mejor relaciona a la herramienta BI es un modelo del tipo constelación.

3.4 Diseño del ETL

Para comenzar con el diseño del proceso de ETL es necesario conocer los datos de origen; cabe aclarar que la información interna de la compañía del volumen *sell in* y *sell out* se puede exportar del sistema en un archivo Excel para cada mercado. Desde Bolivia, Paraguay, y Uruguay la información de *Retail Audit* es enviada por las compañías externas en un archivo Excel desde cada mercado; desde los mercados de Chile y Perú se envía una base en formato *QlikView* el cual solo se puede ver mediante dicho programa pero que también es exportable en un archivo Excel. Y por último el mercado de Argentina, es enviada en una base de *Nielsen Answers* la cual también se puede ver solo en dicho programa pero que también es exportable en un archivo Excel.



En cuanto a la información de la tabla de hechos de “volumen KKAA” en cada mercado se consolidan es un archivo Excel con los datos de todas las cadenas bencineras y es enviada al área de SPI.

Debido a que todos los datos de origen pueden obtenerse en formato Excel, la plataforma para ETL será en Excel programada en VBA y la herramienta encargada es llamada “MDB Administration”.

El “MDB Administration” utilizado por el equipo P&I, es la herramienta utilizada para agregar la información a la base de datos del *software*; luego de ser verificada su veracidad y chequear que concuerde con lo observado en el MDB Information, es liberada para la visualización del resto de las áreas de marketing. Este proceso, el cual posee una gran importancia, es donde se produce la mayor cantidad de errores siendo los más comunes la duplicidad en los datos, carga de campos vacíos y totales erróneos debido a productos que son renombrados en la extracción de la información de origen y no son clasificados en la base de datos, además en las tablas se crearán *Primary Key* de tal manera que se reafirme la unicidad de datos.

La validación previa a la carga de la información en la base de datos, es muy necesaria para evitar los errores nombrados previamente. Por ejemplo, al ingresar el volumen de un producto determinado, que no se encuentra creado en las tablas de relaciones (Relación RA, Relación KKAA y Relación Vol), no se verá reflejado en las *pivot* o reportes del MDB Information.



3.5 Diseño de la interfaz de usuario

Las interfaces básicas de usuario son aquellas que incluyen elementos como menús, ventanas, contenido gráfico, cursor, *beeps* y algunos otros sonidos que el computador emite, y en general, todos aquellos canales por los cuales se permite la comunicación.

El objetivo del diseño de una interfaz es producir una interfaz que sea fácil de usar, eficiente y agradable para que al operar la herramienta dé el resultado deseado. Para ello se comenzará con utilizar como base la interfaz de Microsoft Excel con la cual se encuentran muy familiarizados todos en el Departamento de Marketing de la organización en estudio. Además, para el diseño de las pestañas de la herramienta se utilizarán como premisa las heurísticas de Nielsen.

Basándose en estas heurísticas, en las secciones siguientes se detallarán las interfaces creadas para la herramienta iniciando con “MDB Information” y seguido del “MDB Administration”. También se debe considerar que los principales usuarios son personas jóvenes las cuales están muy familiarizadas con las páginas web. Por lo que el usuario hará un reconocimiento de la interfaz de arriba hacia abajo y de izquierda a derecha, donde lo que tenga un carácter de mayor importancia se situará arriba a la izquierda y lo de menor importancia se situará abajo a la derecha.

3.5.1 Interfaz de usuario de “MDB Information”

Al abrir el “MDB Information” se visualizará un menú Principal como se muestra en la figura 13.



Figura 14: Menú Principal

El lenguaje utilizado para nombrar cada *link* del menú principal fue basado en la segunda heurística de Nielsen nombrada en la sección 2.8, la cual describe que un sistema debe relacionarse con el usuario a través de un lenguaje familiar, el cual utilice a diario. De esta forma, cada *link* tenga un sentido lógico.

Los *links* que se sitúan al lado izquierdo dentro de la interfaz del menú principal apuntan a hojas dentro del mismo archivo Excel donde se han creado previamente tablas dinámicas. Cabe mencionar que las tablas dinámicas son una herramienta que se utiliza a diario en el Departamento de Marketing la cual ya tiene una interfaz definida; sin embargo, se adicionó un botón Volver (ver figura A-1 del anexo), para poder retroceder al menú principal en caso de tener que consultar otro tipo de información, esto es basado en la tercera heurística de Nielsen la cual dice que el usuario debe tener control y libertad dentro del sistema apoyando funciones de deshacer y rehacer.



Los *links* dentro de la interfaz del menú principal situados a la derecha direccionan a hojas modelo, las cuales tienen un diseño predefinido ya que son formatos de reportes automatizados que son entregados por el área periódicamente como se puede ver en la Figura 14.

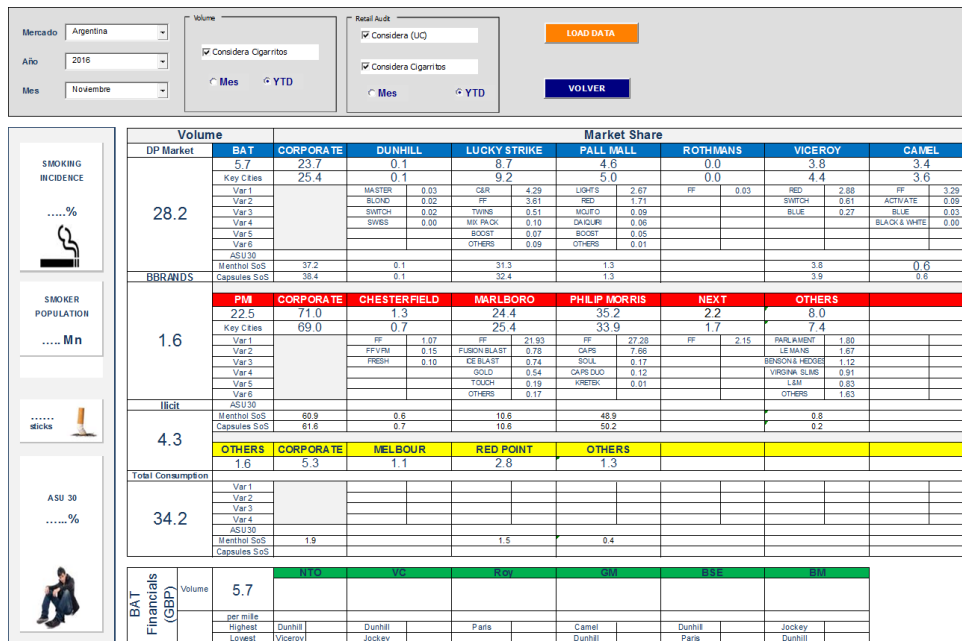


Figura 15: Ejemplo de Reporte Automatizado de la participación de mercado por marca

En estas hojas de reporte requieren que se completen ciertos campos para que pueda ser ejecutado el autocompletado del reporte. Estos campos se encuentran situados en la parte superior de la hoja, donde a través de *Combo Box*, *Check Box* y *Option Button* se selecciona la información deseada. En caso de ejecutar la automatización sin haber completado todos los campos obligatorios, aparecerán mensajes de error indicando cual es la falta y la acción que seguir para corregirla. Todo esto se encuentra basado principalmente en las heurísticas 2, 3, 5, 6 y 8 de Nielsen, donde en breves palabras lo que se propone con esta interfaz es mantener un lenguaje cotidiano, otorgarle al usuario



control y libertad, mantener visibilidad de los objetos, acciones y opciones, y por último mostrar un diseño minimalista donde solo se visibilice la información relevante.

3.5.2 Interfaz de usuario de “MDB Administration”

En el menú de administración de los datos (ver figura A-2 del anexo), se pueden observar las diferentes funciones que posee, como carga de datos, borrado de datos, edición de datos, y habilitación de *links* en el “MDB Information”.

Los *links* que se observan a la izquierda dentro del menú permiten acceder a las principales interfaces para la manipulación de los datos; estos se situaron en dicho lugar debido a que al ser usados repetidas veces tienen un alto grado de importancia. Los botones situados a la derecha son utilizados con menor frecuencia; también basándose en la segunda heurística de Nielsen los nombres de los *links* mantuvieron los de la información manejada previamente por la área, lo cual proporciona un orden natural y lógico en la relación entre el sistema y el mundo real. Estos *links* conducen a hojas de Excel que indican qué datos completar de cada columna (ver figura A-3 del anexo), basándose en la sexta heurística de Nielsen que indica que el usuario debe reconocer antes que recordar, dado que estas interfaces están creadas de tal manera que simulen el mismo orden que se utilizaría para crear una tabla dinámica con datos que se encuentren en el mismo archivo, esto es debido a que posee con anterioridad la costumbre de hacer tablas dinámicas con esa estructura de datos. Estas hojas poseen un botón verificador previo al botón de carga, el cual chequea que la información sea consistente, es decir que no tenga duplicidades ni campos vacíos en las claves primarias. En el caso de



existir algún error en la verificación de los datos, el sistema arrojará una alerta informando cual y donde se sitúa el error.

Los botones que se encuentran a la derecha del menú (ver figura A-2 del anexo), dirigen a otras ventanas de sub-menús respectivos de cada botón (ver figura A-4 del anexo), donde se visualizarán de la misma forma que en el menú principal las acciones que se deseen ejecutar.

Al seleccionar alguna de las acciones, ya sea agregar, editar o borrar algún dato de la base, se abrirá una ventana de selección como se puede apreciar en la figura 15.

Maestro SKU codigo SAP

Agregar M SKU

SKU

COD_END_MARKET:

COD_SKU_SAP:

COD_SKU_SYSTEM:

COMPANY:

PRODUCT_DESC_STANDARD:

PRODUCT_DESCRIPTION:

LENGTH:

VERSION:

TUBE:

PRODUCT_LINE:

FLAVOUR:

GDB:

IB:

STRENGTH:

CIRCUNFERENCE_TYPE:

PRICE_SEG_WAP:

BRAND_HOUSE:

PACK_TYPE:

QTY_STICKS:

FILTER_TYPE:

CAPSULE:

CAPSULE_FLAVOUR_TYPE:

QTY_CAPSULE:

DUTY_FREE:

KB:

ADDITIVE_FREE:

Volver

Agregar SKU

Figura 16: Ejemplo de ventana de una selección de acción agregar SKU

Todas las ventanas de selección poseen la misma interfaz, con la salvedad que se modifica el tamaño dependiendo de la cantidad de datos que se necesiten para ejecutar la acción. Los datos se solicitarán a través de *Combo Box*, *Check Box* y *text box*, pero



también en las acciones de agregado y edición de los datos estará la función de autocompletado. Esta función se activa al completar los pasos que se le indicarán al usuario en la interfaz resaltándose de un color distinto cada paso a seguir para que no se desvíe la atención del usuario. En el caso de completar manualmente los datos que se desee ingresar a la base y el usuario cometa algún error, el sistema también proporcionará un aviso de advertencia indicando cual ha sido el error y el paso a seguir para solucionarlo. La acción de borrado de datos se diferencia de las acciones de agregado y edición, ya que en ella no es necesario completar todos los campos que figuran en la ventana. Esto es debido a que cada dato que se solicita actúa como un filtro en el borrado: mientras más casillas se completen más preciso será el borrado. Siempre que se ejecute una acción de borrado el sistema hará aparecer una ventana de reconfirmación, asegurando que lo seleccionado por el usuario sea lo correcto.

3.6 Indicador clave del desempeño de la herramienta

Los KPI o indicadores clave de desempeño son mediciones financieras o no financieras, que se utilizan para medir el grado de cumplimiento de los objetivos previamente establecidos en una empresa o proyecto.

En el caso de una empresa, los KPIs se utilizan para determinar el estado actual de un negocio y permiten definir una línea de acción futura; expresan las variables a tomar en cuenta para incrementar el desempeño favorable para conseguir el progreso, posicionamiento y ganancias significativas de la empresa. El KPI mide las condiciones para el rendimiento y el desempeño de manera clave (22).



Para el presente proyecto se establecieron indicadores o métricas cuantificadas, alineadas a los objetivos planteados en la sección 1.5, las cuales se consideraron como los KPI. Estos se utilizaron durante el desarrollo de la herramienta, los cuales ayudaron a mantener bajo control el cumplimiento de los objetivos y eliminar las causas de los defectos en las diferentes pruebas del ciclo de vida del desarrollo de la herramienta.

Se utilizaron principalmente dos KPI para evaluar el desempeño de la herramienta MDB descrita en este trabajo, los cuales son: facilidad de uso y tiempo de espera en la ejecución de los reportes.

Para la facilidad de uso, la herramienta se creó basándose en las heurísticas de Nielsen como se explicó en la sección 3.5. Para poder evaluar este KPI, se midió el tiempo en que el usuario lograba encontrar la información que buscaba y, además, si el usuario recordaba en una segunda evaluación donde había encontrado la información.

El segundo KPI que se utilizó para esta herramienta fue el tiempo que se demoraron los usuarios en armar correctamente un informe oficial. Este indicador también ayudó a generar una alerta en caso de que el tiempo de los reportes tome mucho tiempo en ejecutarse, señal de que la herramienta comenzó a quedar obsoleta.

En el transcurso del desarrollo de la herramienta, estos KPI fueron cubiertos satisfactoriamente. Es decir que al evaluar el MDB, el KPI de facilidad de uso indicó que ningún usuario tomó más de 1 minuto en obtener la información que necesitaba por lo cual se considera satisfactoriamente cumplida la evaluación de la interfaz.



Por otro lado, al evaluar el segundo KPI, resaltó el hecho que el tiempo de espera por cada reporte variaba según su complejidad y el mercado que se escogiera como filtro para poder ser creado; sin embargo, la ganancia en tiempo es considerable debido a que los mismos usuarios estimaron que el armado de un reporte tomó un 10% del tiempo en el cual se construía anteriormente el mismo reporte.



Capítulo 4: Validación de la Solución

En este capítulo se mostrarán, las tareas a ejecutar por los actores a modo de prueba para corroborar la calidad y la existencia de algún posible error en el sistema.

4.1. Test de Aceptación

En el contexto del desarrollo de un proyecto de *software*, un apropiado *testing* es fundamental para el éxito de la fase de producción. En este sentido, el *test* de aceptación puede definir las pruebas que el *software* debe superar para ser aceptado y comenzar su funcionamiento.

Para describir el *test* de aceptación utilizado, se mencionarán las sesiones y sus correspondientes listas de pruebas a superar. También se describirán las fallas dentro de las sesiones y sus causas, las cuales han sido resueltas en los reportes de errores.

4.1.1. Primera sesión

En la primera sesión del test de aceptación se plantearon las siguientes tareas a realizar con el sistema.

- a) Registrar usuarios: esta tarea consiste en poder ingresar un usuario nuevo al sistema completando los datos requeridos, para poder corroborar que la interface sea intuitiva.
- b) Registrar datos en las tablas principales: esta acción consiste en efectuar una carga inicial de datos a la base, de tal manera que se puedan corroborar las alertas de posibles fallas. El usuario debe entrar a cualquiera de las cargas de



volumen, copiar la información, efectuar la validación que la data se encuentre correcta y finalizar con la carga en la base.

- c) Registrar Productos: el usuario trata de cargar datos en las tablas principales, pero aparece una ventana que le alerta sobre una falla, la cual identifica la ausencia de un SKU y le indica que debe ingresar el SKU al sistema para evitar el error
- d) Consultar el ingreso de usuario diferenciado: distintos usuarios ingresan al sistema corroborando que quede un registro de quien ingresó, a qué hora ingreso, si estaba autorizado a ver todos los *links* y qué *links* utilizó del sistema. De esta manera se llevará un registro de uso del sistema, observando las horas de mayor uso, que área es la que lo consulta con mayor frecuencia, y mantener la privacidad de la información.
- e) Consultar tablas dinámicas, los resultados y la flexibilidad de la manipulación de estos: esta tarea consiste en corroborar que las tablas dinámicas contienen toda la información que fue solicitada para los análisis del área.
- f) Ejecutar reportes automatizados: esto consiste en verificar que los cálculos y el auto completado de los reportes sean correctos.

4.1.2. Primer reporte de errores

Los errores que se presentaron luego de realizar las pruebas de la subsección fueron corregidos y complementados con observaciones hechas durante el proceso. Estas correcciones se presentan a continuación:



- a) Corrección de ingresos de datos vacíos: el sistema no efectuó ningún tipo de alerta cuando se intentó cargar celdas vacías a las tablas de volumen principales.
- b) Tener un registro de la fecha en la cual se ingresaron datos, quién y qué tipo de datos.
- c) Ciertas tablas dinámicas que permanezcan fijas evitando la flexibilidad de la información, debido a que no todos los usuarios poseen el mismo conocimiento para la manipulación de estas, evitando el mal entendimiento del dato observado en estas.
- d) En reportes personalizados, mantener un orden general donde primero se debe observar los datos de la compañía, luego los de la competencia internacional y finalmente agrupar el resto de la competencia en cada mercado.
- e) En los reportes personalizados, colocar agregar filtros de fecha que permitan ver reportes previos al actual.
- f) Nuevos reportes y tablas dinámicas.

Luego de haber corregido los errores en el código, a través de alertas y pequeñas modificaciones en el código, se prosiguió a efectuar las modificaciones solicitadas por los usuarios en los reportes personalizados. A continuación de la corrección se efectuó una segunda sesión de validación de la herramienta.



4.1.3. Segunda sesión

En la segunda sesión del *test* de aceptación se corroboraron las correcciones y se realizaron pruebas de los comentarios realizados en la sesión previa que complementarían el sistema. A continuación, se presenta una lista con las pruebas a corroborar:

- a) Corrección del ingreso de información vacía.
- b) Registro de los usuarios que modificaron de alguna forma la base de datos ya sea por insertar, modificar o borrar.
- c) Orden de los reportes y sus respectivos cálculos.
- d) Nuevos reportes y tablas dinámicas

4.1.4. Segundo reporte de errores

Los errores y comentarios que se presentaron luego de realizar las pruebas anteriores son los siguientes:

- a) Las variaciones porcentuales de los reportes deben indicar “pp”.
- b) Bloquear ciertos accesos a las tablas dinámicas cuando se estén efectuando cargas.
- c) Enviar un mail masivo a todos los usuarios cuando las tablas dinámicas estén habilitadas.



Luego de haber efectuado las correcciones de la segunda sesión de evaluación de la herramienta, se dio por finalizado el proyecto.



Capítulo 5: Conclusiones

El ambiente del mundo de los negocios de hoy exige una aplicación cada vez más eficiente sobre la información disponible. Las herramientas BI generan el conocimiento al negocio, que se deriva de la correcta utilización de la información generada dentro y fuera de la empresa.

La herramienta MDB creada en el presente trabajo pone a disposición de los usuarios la información correcta en el lugar correcto, con múltiples beneficios para la empresa, entre ellos la generación de una ventaja competitiva.

Se puede observar que se pudieron cumplir los objetivos generales y específicos, con la herramienta BI que se proporcionó al área de SPI. Ya que pudo mejorar el desempeño, disminuyendo el tiempo de manipulación de la información, y proporcionando mayor tiempo para análisis y otras funciones de la área, lo que provocó una mejora en la competitividad de la área. Por otra parte, también se logró comprender los procesos en la gestión del área SPI, con lo que se pudo modelar y crear una base de datos que pudiese reunir la información necesaria para el desempeño de las principales funciones del área, y generar la posibilidad de nuevas comparaciones entre las fuentes de información, como por ejemplo analizar las tendencias del volumen *Sell In* con el volumen *Sell Out* y el volumen RA. También se estableció una interfaz fácil de usar, ya que se implementó el *software* Excel como plataforma, que es familiar por su utilización en las tareas diarias del área.



En el transcurso de la elaboración del sistema MDB, cabe mencionar que no es estrictamente necesario que el formato de los reportes personalizados se encuentre en formato Excel, el cual es el formato que siempre se ha trabajado; debido a esto, es posible proponer a modo de mejora continua del sistema, que los reportes personalizados sean migrados a una plataforma distinta donde los tiempos de respuesta sean mejores. Esta propuesta es debido a que los reportes personalizados son reportes con una estructura rígida, la cual no necesita la flexibilidad de manipulación de datos que requiere las tablas dinámicas las cuales están más enfocadas al análisis.

Dentro de las fases de ETL en los procesos de extracción de la información se tuvo que mantener los procesos de recaudación de la información manualmente, debido a que existen 10 fuentes de datos de origen las cuales todas poseen formatos particulares y que varían en el tiempo, por lo que actualmente es imposible efectuar alguna automatización en la extracción de la información. Sin embargo, queda abierta la posibilidad de una mejora en la automatización de la extracción de la información de origen.

La exigencia de una herramienta BI en las distintas áreas o departamentos de una empresa multinacional es imperiosamente necesaria, debido a que los grandes sistemas BI captan mucha información innecesaria, dependiendo del área que los utilice. Por lo cual, podemos concluir, que la utilización de una herramienta BI que centralice de manera local la información necesaria de cada área, sin necesidad de acudir a estos sistemas globales, optimiza los tiempos de los procesos internos de cada área. Pero gracias al presente trabajo se puede afirmar que la factibilidad de una herramienta BI generalizada que pueda implementarse de manera centralizada y local de igual manera



en cualquier área no es posible, debido a que cada área tiene sus propias funciones, sus propias fuentes de información y sobre todo sus propios procesos internos; es por esto que cada herramienta BI debe ser personalizada por las necesidades que requiere cada área.

Finalmente cabe resaltar que en el desarrollo de la solución BI creada para la área SPI fue muy restringido en el aspecto técnico, debido a que las herramientas de programación que poseía la empresa en el Departamento de Marketing eran extremadamente limitadas; es por esta razón que se utilizó el lenguaje VBA y la base de datos *Access*. Esta limitante en la programación, deja una gran oportunidad para el mejoramiento de la herramienta, dando la posibilidad de cambiar el entorno de la programación y de la base de datos, de tal manera que se mejoren los tiempos de ejecución en las consultas y ampliando las posibilidades de efectuar reportes automatizados de mayor complejidad.

La aplicación MDB que fue creada y presentada como una herramienta BI para solucionar y apoyar el Departamento de Marketing, para poder prolongar su eficiencia en el tiempo, es recomendable trazar una proyección de escalabilidad del tipo vertical. El escalado incluye agregar más memoria, más procesadores o procesadores más rápidos o, simplemente, migrar la aplicación a un equipo más potente. Normalmente, este trabajo permite un aumento en la capacidad sin requerir cambios en el código fuente.

El incremento que posee la base de la herramienta BI propuesta generó una mayor demanda de recursos. Además, por el hecho que el Departamento de Marketing no



posee el conocimiento de programación para modificar de alguna forma el código fuente, hace muy dudoso que se pueda implementar otro tipo de una escalabilidad.

Es por eso que la mejor manera de proyectar un buen desempeño y durabilidad de la herramienta es mejorando hardware para poder entregar mayores recursos de procesamiento al sistema.

Por otra parte, debe ser considerada la posibilidad de poder migrar la base de datos a un servidor local y no situarla en el *cluster* de la empresa en Brasil, eso mejoraría la conectividad a la base de datos por lo que agilizaría la herramienta creada.



Bibliografía

1. BAT Chile. [En línea] 1-
http://www.batchile.com/group/sites/BAT_9YFD2P.nsf/vwPagesWebLive/DO9T5K3S?opendocument.
2. BAT. [En línea] 2-
http://www.batcolombia.com/group/sites/bat_87af9v.nsf/vwPagesWebLive/DO87XHEH?opendocument.
3. Ranjan, Jayanthi. BUSINESS INTELLIGENCE: CONCEPTS, COMPONENTS, TECHNIQUES AND BENEFITS. *Journal of Theoretical and Applied Information Technology*. 2009.
4. Rubio, Gerardo. *The 5 Styles of Business Intelligence*. Microstrategy Inc. Mexico : s.n., 2001.
5. Wikipedia La enciclopedia libre. [En línea] Wikipedia, Inc, 12 de 2 de 2017. [Citado el: 20 de 5 de 2017.] https://es.wikipedia.org/wiki/Herramientas_de_inteligencia_de_negocios.
6. Espinosa, Roberto. Rincon de BI. [En línea] Abril de 2010.
<https://churriwifi.wordpress.com/2010/04/19/15-2-ampliacion-conceptos-del-modelado-dimensional/>.
7. R.rivadero, Gustavo. Metodología Kimball. [En línea] 2010.
<http://www.interaktiv.cl/blog/wp-content/uploads/2012/04/MetodologiaKimball.pdf>.
8. DataPrix. *Bernabeu R. Dario*. [En línea] 6 de 5 de 2009.
<http://www.dataprix.com/datawarehouse-manager>.
9. M., Lucia I. Cardoso. *Sistema de Base de Datos 2*. 2006. pág. 137.
10. Szepekuti, Istvan. *Multidimensional or Relational ?* Budapest : Nationale-Nederlanden Hungary Insurance Co. Ltd., 2000.
11. Codd, E.F., S. B. Cood, and C.T. Selley. *Providing OLAP (On-line Analytical Processing) to User-Analysts: An IT Mandate*. s.l. : E.F. Cood & Associates., 1993.
12. Pendse, Nigel. *"What is OLAP"*. 1998.
13. Lopez, Darien Montes. Topicoss. [En línea] 19 de Mayo de 2010.
<https://topicoss.wordpress.com/>.
14. Encinas, Maria Trinidad Serna. 1ºer. Taller de investigación Base de datos multidimensionales. 2016. 23.
15. Sinnexus. *Business Intelligence Informativa estrategica*. [En línea]
http://www.sinnexus.com/business_intelligence/olap_avanzado.aspx.



16. Rivas, Antonio. Aprendiendo Business Intelligence. [En línea] Febrero de 2011.
<http://www.bi.dev42.es/2011/02/23/olap-molap-rolap/>.
17. SAS The power to know. [En línea] [Citado el: 28 de 9 de 2017.]
https://www.sas.com/en_us/insights/data-management/what-is-etl.html.
18. Espinosa, Roberto. DataPrix. [En línea] [Citado el: 28 de 9 de 2017.]
<http://www.dataprix.com/blogs/respinosamilla/herramientas-etl-que-son-para-que-valen-productos-mas-conocidos-etl-s-open-sour>.
19. Software, Laboratorio Nacional de Calidad del. Instituto Nacional de Tecnologías de. España : Ingeniería del Software: Metodologías y Ciclos de Vida, 2009, págs. 24 - 36.
20. Ciclo de vida del software. *Implementacion y Debugging*. 1.
21. Finelli, Fernando. Braintive. [En línea] [Citado el: 20 de 9 de 2017.]
<http://www.braintive.com/10-reglas-heuristicas-de-usabilidad-de-jakob-nielsen/>.
22. Cano, Carlos. Que es un KPI. *Comenzando de cero*. [En línea]
<https://comenzandodecero.com/que-es-un-kpi/>.
23. Engineers., Institute of Electrical and Electronics. IEEE Standard Glossary of Software. 1990, págs. 1 - 84.

Anexo

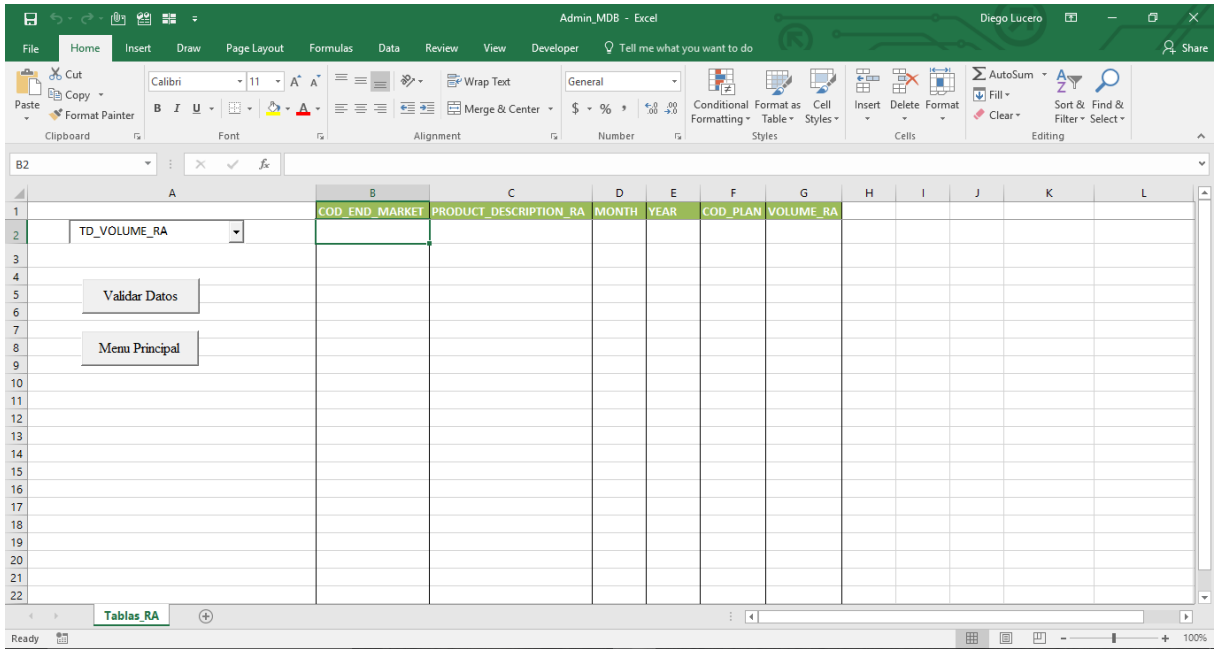
The screenshot shows an Excel spreadsheet with a dynamic table titled 'SHIPMENTS'. The table has columns for various attributes like STRENGTH, FILTER_TYPE, TUBE, DUTY_FREE, PACK_TYPE, PRICE_SEQ_WAP, LENGTH, CIRCUMFERENCE_TYPE, CAPSULE_FLAVOUR_TYPE, GDB, and PRODUCT_DESC_STANDARD. There are also columns for COD_PLAN, ACTUAL, QTY_STICKS, BRAND_HOUSE, PRODUCT_DESC, COD_SKU_SAP, CAPSULE, FLAVOUR, QTY_CAPSULES, and PRODUCT_LINE. A 'VOLVER' button is visible. Below the table is a summary table for 'Sum of SumOfVOLUME1' with columns for Year (2016) and Month (1-10). The data table shows values for different companies (BAT, PMI, BBRANDS, UC) and a Grand Total row.

Sum of SumOfVOLUME1		Year	Month										
		2016		1	2	3	4	5	6	7	8	9	10
COD_END_MARKET	COMPANY												
AR	BAT			577,643	596,285	593,175	647,632	369,532	501,537	489,035	575,956	488,256	500,544
	PMI			2,533,436	2,362,989	2,640,515	2,745,401	1,489,119	2,163,625	1,898,967	2,360,355	2,111,879	2,176,700
	BBRANDS			138,252	154,685	168,518	151,885	114,760	150,058	156,413	170,521	203,143	181,598
	UC			412,464	405,912	448,916	472,528	265,772	459,210	415,125	505,867	456,712	475,340
Grand Total				3,661,795	3,569,871	3,849,125	4,017,445	2,239,183	3,274,431	2,969,539	3,612,699	3,269,983	3,334,182

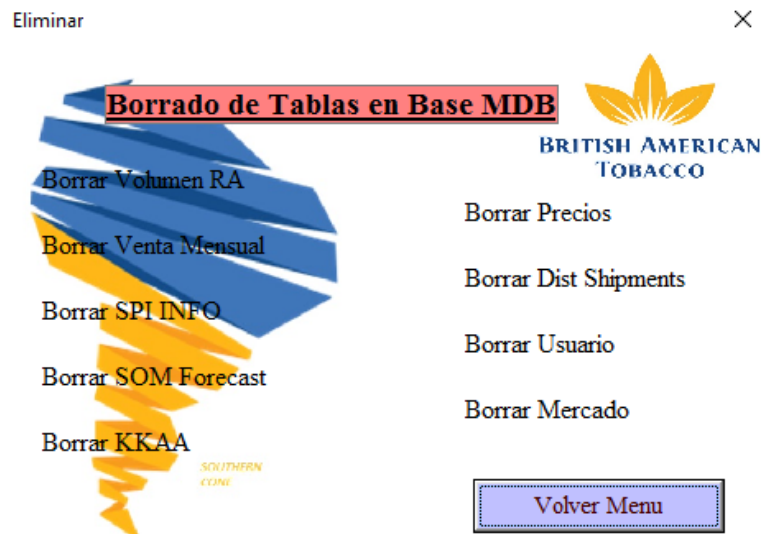
A - 1: Ejemplo de Hoja con Tabla Dinámica

The screenshot shows a web application interface titled 'Menu Administrador'. It features a logo for 'BRITISH AMERICAN TOBACCO' and a list of menu items on the left side, including 'Carga Tablas RA', 'Carga Venta Mensual', 'Cargas KKA', 'Carga SPI INFO', 'Carga Precio', 'Carga Dist Shipments', 'Carga SOM Forecast', and 'Carga L4L'. On the right side, there are several action buttons: 'Borrar de la base', 'Agregar a Maestros', 'Update Maestros', 'Habilitar Links MDB', and 'Salir Admin MDB'.

A - 2: Menú MDB Administration



A - 3: Template de Carga



A - 4: Ejemplo ventaja sub-menu