



Pontificia Universidad Católica del Ecuador

Facultad Ingeniería

Carrera Sistemas y Computación

**Trabajo previo a la obtención del título de
Ingeniero en Sistemas y Computación**

**Análisis Comparativo de Migración de Motor de
BDD de SQL Server a Oracle en una Empresa del
Sector Automotriz.**

Autor: Gabriel Montalvo

Director: Master Javier Cóndor Cruz

2019 - JUL

Tabla de contenido

1.	INTRODUCCIÓN.....	1
2.1.	Sistemas Gestores de Bases de Datos DBMS.....	3
2.1.2.	Modelo entidad-relación.....	5
2.1.3.	Modelo relacional.....	6
2.2.	DBMS Sql.....	8
2.2.1.	Oracle Database 11g.....	8
2.2.2.	SQL Server 2012.....	10
2.2.3.	Otros DBMS Sql.....	12
2.3.	DBMS No Sql.....	13
2.3.1.	MongoDB.....	13
2.3.2.	Amazon DynamoDB.....	14
2.4.	Definición de Data.....	14
2.4.1.	DATA en una Base de Datos Relacional.....	15
2.5.	Migración de Datos.....	17
2.5.1.	Tipos de Migración.....	17
2.6.	Definición de ERP.....	19
2.6.1.	Ventajas de un ERP.....	19
2.6.2.	Desventajas de un sistema ERP.....	20
3.	CAPÍTULO II: Situación Actual de Proauto C.A.	21
3.1.	Historia.....	21
3.2.	Líneas de Negocio.....	22
3.2.1.	Comercialización de Vehículos.....	22
3.2.2.	Servicio de Talleres.....	23
3.2.3.	Venta de Repuestos.....	24
3.3.	Organigrama de la Empresa.....	24
3.4.	Situación Financiera.....	25
3.5.	Procesos del Negocio.....	27
4.	CAPÍTULO III: ERP DMS.....	29
4.1.	Generalidades.....	30
4.2.	Infraestructura.....	31
4.3.	DBMS con el que Trabaja.....	32
4.3.1.	Modelo Entidad Relación en SQL Server.....	33

5.	CAPÍTULO IV: ERP KAIROS.....	43
5.1.	Generalidades.....	43
5.2.	Infraestructura	45
5.3.	DBMS con el que Trabaja.....	46
5.3.1.	Modelo Entidad Relación en ORACLE	46
6.	CAPÍTULO V: MIGRACION DE DATA.....	56
6.1.	Antecedentes	56
6.2.	Secuencia Lógica de Migración.....	57
6.2.1	Migración de data en Caso de estudio sin aplicar secuencia Lógica.....	62
6.3.	Análisis de DATA.....	62
6.4.	Determinación de ETL.....	69
6.5.	Pruebas de calidad de Datos	72
7.	CAPÍTULO VI: CONCLUSIONES Y RECOMENDACIONES	74
7.1.	Conclusiones	74
7.2.	Recomendaciones	75
	BIBLIOGRAFIA	78
	ANEXOS	80
	Anexo 1: Organigrama General de Proauto.pdf	80
	Anexo2: Diagrama de Procesos del área Comercial.	80
	Anexo3: Diagrama de Procesos de Talleres.	80
	GLOSARIO	81

1. INTRODUCCIÓN

En la actualidad, las empresas tienen la necesidad de poder administrar grandes cantidades de información; información que es generada a diario por cada uno de los usuarios que interactúan con los distintos aplicativos que manejan. El sector automotriz siempre está en búsqueda de la automatización de sus procesos y mejorar la tecnología para optimizar sus operaciones.

Por esta razón, se presentan con frecuencia proyectos de migración de datos entre distintos DBMS. Además, no siempre se puede contar con todos los recursos para hacerlo de una manera ordenada, así como el tiempo suficiente o una metodología adecuada para el desarrollo del proyecto.

Es así como PROAUTO C.A., empresa dedicada a la venta y postventa de vehículos de la marca Chevrolet, realizó entre octubre del 2016 y marzo del 2017 el cambio de ERP; entre un sistema con SQL Server a un sistema con gestor de base de datos Oracle 11g. Este cambio generó una migración de datos entre los dos motores de base de datos, con la premisa de realizarlo en 3 meses.

Basado en esta migración para ser implementada en tiempo record; se planteará en este trabajo una metodología que permita el paso ordenado de la data entre los dos gestores de base de datos. Además de un análisis de los tipos de datos y los modelos entidad relación que maneja cada ERP.

En resumen, los capítulos llevarán el siguiente contenido:

Capítulo 1, conceptos de sistemas gestores de bases de datos, migración de data y otros conceptos teóricos importantes.

Capítulo 2, información más relevante de la empresa PROAUTO C.A. así como sus procesos, historia y demás información relevante.

Capítulo 3, análisis del sistema ERP DMS con una visión hacia sus datos.

Capítulo 4, análisis de KAIROS ERP orientado a revisar sus estructuras de datos.

Capitulo 5, migración de data y metodología propuesta de migración

Capitulo 6, conclusiones y recomendaciones.

Palabras Clave: Migración, Base de Datos, Oracle, SQL Server, ERP, Data, DBMS.

2. CAPÍTULO I: Marco Teórico

En este capítulo se va a revisar todos los conceptos teóricos para sustentar el desarrollo de este trabajo. De manera general se detallará los conceptos de Sistemas Gestores de Bases de Datos, y de la Data que administran. Así como también los diferentes Gestores Sql y los No Sql. Por otro lado, se incluirá información acerca de migración y una revisión rápida de los conceptos, ventajas y desventajas de un ERP¹.

2.1. Sistemas Gestores de Bases de Datos DBMS²

A lo largo de las últimas cuatro décadas del siglo veinte y las dos primeras del siglo veinte y uno, el uso de las bases de datos creció en todas las empresas. A sus inicios, muy pocas personas interactuaron directamente con los sistemas de bases de datos, aunque sin darse cuenta interactuaron con bases de datos indirectamente (con los informes impresos como extractos de tarjetas de crédito, o mediante agentes como cajeros de bancos y agentes de reserva de líneas aéreas). Después vinieron los cajeros automáticos y permitieron a los usuarios interactuar con las bases de datos. (Abraham Silberschatz, 2002)

Las interfaces telefónicas con los computadores (sistemas de respuesta vocal interactiva) también permitieron a los usuarios manejar directamente las bases de datos. Un usuario, podía marcar un número y pulsar teclas del teléfono para introducir información o para seleccionar opciones, para determinar las horas de llegada o salida, por ejemplo, o para matricularse de asignaturas en una universidad. (Abraham Silberschatz, 2002)

¹ ERP: sistemas de planificación de recursos empresariales en inglés, enterprise resource planning

² DBMS: (Data Base Management System). Son las siglas en inglés para los Sistemas de Gestión...

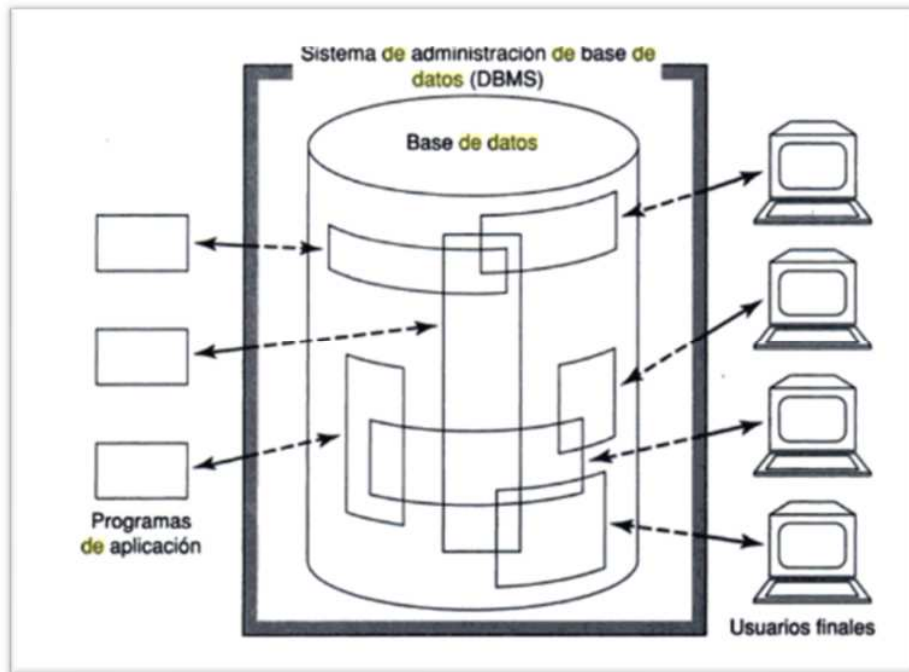


Figura 1 -1 Sistema de Gestión de Bases de Datos (DATE, 2001)

La Figura 1-1 nos permite ilustrar como un Sistema Gestor de Base de Datos está formado por 4 elementos principales: Software, Hardware, datos y usuarios respectivamente. Siendo considerado como Software los Programas de aplicación, Hardware los PCs, y servidores, Data almacenada en la Base de Datos y por último los usuarios finales.

2.1.1. Visión de los Datos

Las bases de datos están compuestas, de datos y de metadatos. Los metadatos son datos que sirven para especificar la estructura de la base de datos; por ejemplo, qué tipo de datos se almacenan (si son texto o números o fechas), qué nombre se le da a cada dato (nombre, apellidos, etc), cómo están agrupados, cómo se relacionan. (Sanchez, 2004)

De este modo se producen dos visiones de la base de datos:

- **Estructura lógica.** Indica la composición y distribución teórica de la base de datos. La estructura lógica sirve para que las aplicaciones puedan utilizar los elementos de la base de datos sin saber realmente cómo se están almacenando. Es una estructura que permite idealizar a la base de datos. Sus elementos son objetos, entidades, nodos, relaciones, enlaces, que realmente no tienen presencia real en la

física del sistema. Por ello para acceder a los datos tiene que haber una posibilidad de traducir la estructura lógica en la estructura física. (Sanchez, 2004)

- **Estructura física.** Es la estructura de los datos tal cual se almacenan en las unidades de disco. La correspondencia entre la estructura lógica y la física se almacena en la base de datos (en los metadatos). (Sanchez, 2004)
- **Nivel de vistas:** El nivel más alto de abstracción describe sólo parte de la base de datos completa. Muchos usuarios del sistema de base de datos no necesitan toda esta información. En su lugar, tales usuarios necesitan acceder sólo a una parte de la base de datos. Para que su interacción con el sistema se simplifique, se define la abstracción del nivel de vistas. En el nivel de vistas, los usuarios de computadoras ven un conjunto de programas de aplicación que esconden los detalles de los tipos de datos. Análogamente, en el nivel de vistas se definen varias vistas de una base de datos y los usuarios de esta ven única y exclusivamente esas vistas. Además de esconder detalles del nivel lógico de la base de datos, las vistas también proporcionan un mecanismo de seguridad para evitar que los usuarios accedan a ciertas partes de la base de datos. Por ejemplo, los cajeros de un banco ven únicamente la parte de la base de datos que tiene información de cuentas de clientes; no pueden acceder a la información referente a los sueldos de los empleados. (Yera, 2007)

2.1.2. Modelo entidad-relación

Un diagrama entidad-relación, también conocido como modelo entidad relación o ERD, es un tipo de diagrama de flujo que ilustra las "entidades", como personas, objetos o conceptos, se relacionan entre sí dentro de un sistema. Los diagramas ER se usan a menudo para diseñar o depurar bases de datos relacionales en los campos de ingeniería de software, sistemas de información empresarial, educación e investigación. También conocidos como los ERD o modelos ER, emplean un conjunto definido de símbolos, tales como rectángulos, diamantes, óvalos y líneas de conexión para representar la interconexión que existe entre entidades, relaciones y sus atributos. Son un reflejo de la estructura gramatical y emplean entidades como sustantivos y relaciones como verbos. (Lucidchart, 2018)

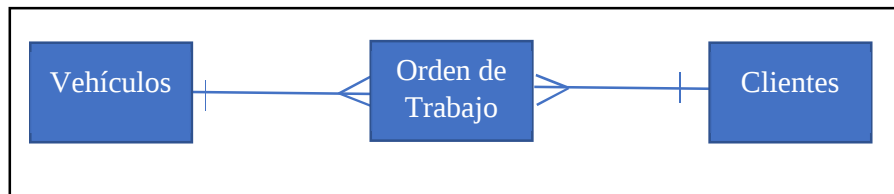


Figura 1- 2 Modelo Entidad Relación (Montalvo, 2019)

La figura 1-2 es la representación gráfica de un modelo entidad relación donde se puede observar que existe una relación entre las entidades vehículos, Orden de Trabajo, Clientes. De manera que se lee que un vehículo puede tener muchas ordenes de trabajo y lo mismo un cliente puede tener muchas ordenes de trabajo, o se podría leer como una Orden puede pertenecer a un solo vehículo o a un solo cliente.

2.1.3. Modelo relacional

Modelo de organización y gestión de bases de datos consistente en el almacenamiento de datos en tablas compuestas por filas, o tuplas, y columnas o campos. Se distingue de otros modelos, como el jerárquico, por ser más comprensible para el usuario inexperto, y por basarse en la lógica de predicados para establecer relaciones entre los distintos datos.

Surge como solución a la creciente variedad de los datos que integran los almacenes de datos y se puede resumir el concepto como una colección de tablas (relaciones). (Cabello, 2010)

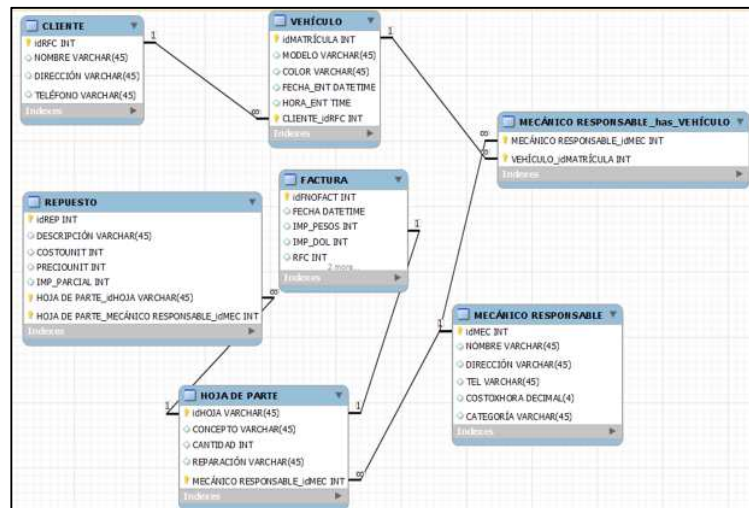


Figura 1-3 Modelo Relacional (Sistema Manejador de Base de Datos (SMBD), 2018)

En la figura 1-3 se puede observar la representación gráfica de un modelo relacional con sus tablas y campos respectivos además de la relación que existe entre cada tabla, también se puede visualizar de manera rápida los tipos de datos que maneja cada campo, como por ejemplo la tabla Cliente tiene su clave principal *idRFC* que es de tipo entero y con tres campos más que son nombre, dirección y teléfono que son de tipo carácter.

2.1.4. Otros Modelos

El modelo de datos orientado a objetos es otro modelo de datos que está recibiendo una atención creciente. El modelo orientado a objetos se puede observar como una extensión del modelo E-R con las nociones de encapsulación, métodos (funciones) e identidad de objeto. El modelo de datos relacional orientado a objetos combina las características del modelo de datos orientado a objetos y el modelo de datos relacional. (Abraham Silberschatz, 2002)

Los modelos de datos semiestructurados permiten la especificación de datos donde los elementos de datos individuales del mismo tipo pueden tener diferentes conjuntos de atributos. Esto es diferente de los modelos de datos mencionados anteriormente, en los que cada elemento de datos de un tipo particular debe tener el mismo conjunto de atributos. El lenguaje de marcas extensible (XML, eXtensible Markup Language) se usa ampliamente para representar datos semiestructurados. (Abraham Silberschatz, 2002)

Históricamente, otros dos modelos de datos, el modelo de datos de red y el modelo de datos jerárquico, precedieron al modelo de datos relacional. Estos modelos estuvieron ligados fuertemente a la implementación subyacente y complicaban la tarea del modelado de datos. Como resultado se usan muy poco actualmente, excepto en el código de bases de datos antiguo que aún está en servicio en algunos lugares. (Abraham Silberschatz, 2002)

2.2. DBMS Sql

Un sistema de gestión de bases de datos relacionales o SQL, es un programa que te permite crear, actualizar y administrar una base de datos relacional. La mayoría de los DBMS Sql comerciales utilizan el lenguaje de consultas estructuradas (SQL) para acceder a la base de datos, aunque SQL fue inventado después del desarrollo del modelo relacional y no es necesario para su uso. (Rouse, 2018)

Los principales DBMS Sql son: Oracle, DB2 de IBM y Microsoft SQL Server de tipo comercial; existen tipo uso libre como MySQL, MariaDB, PostgreSQL. A pesar de los desafíos repetidos por tecnologías de la competencia, así como la afirmación de algunos expertos que dicen que ninguno de los RDBMS actuales ha aplicado plenamente los principios relacionales, la mayoría de las nuevas bases de datos corporativas siguen siendo creadas y gestionadas con un DBMS Sql. (Rouse, 2018)

Hoy en día el mercado de las bases de datos ha crecido mucho, entre bases de datos comerciales y bases de datos de libre distribución, a continuación, se revisarán algunas de ellas. En especial Oracle 11g y SQL Server 2012 que son el sustento teórico de este trabajo de disertación. (Montalvo, 2019)

2.2.1. Oracle Database 11g

La base de datos Oracle es líder en el mercado y representa la base de datos preferida por miles de empresas, desarrolladores y administradores de todo el mundo. Con el transcurso de los años, las empresas han llegado a depender de la base de datos Oracle para brindar confiabilidad y desempeño incomparables. Con la versión 10g, Oracle ofreció una base de datos de autoadministración con capacidades de administración avanzada, reduciendo

drásticamente los costos de administración. Con el lanzamiento de Oracle Database 11g, Oracle está nuevamente preparado para superar los obstáculos. (Athreya, Junio 2007)

Diseñada para entornos de centros de datos en constante cambio y evolución para cubrir las demandas empresariales, Oracle Database 11g permite a las empresas adoptar rápidamente nuevas tecnologías mientras se minimiza el riesgo. Asimismo, sobre la base de sus capacidades de auto administración, Oracle Database 11g ha realizado importantes avances en las áreas de capacidad de administración y diagnóstico de fallas. (Athreya, Junio 2007)

Oracle 11g se comercializa en tres versiones distintas los principios de arquitectura y administración son los mismos en las tres ediciones: Edición Empresarial; Edición Estándar One y Two y Edición Personal.

La Versión Estándar o (Standar Edition) permite o tiene todas las funcionalidades principales que permiten poner en producción las aplicaciones tipo Cliente Servidor orientadas a un grupo de trabajo o departamento de una empresa. Por esta razón Oracle saco al mercado las versiones Standar Edition One y Standar Edition Two, las que permiten trabajar con equipos servidores de mayores prestaciones en cuanto a procesador. (Athreya, Junio 2007)

La Edición Empresarial o (Enterprise Edition) por otro lado está destinada a trabajar con aplicaciones más avanzadas dentro de una organización, posee funcionalidades que permiten mejorar la disponibilidad y capacidad de carga de las grandes bases de datos, al mismo tiempo que facilita su administración y optimización con herramientas propias de la versión como, por ejemplo:

- Oracle Real Application Cluster
- Oracle Partitioning
- Oracle Tuning Pack
- Oracle OLAP³
- Advance Compression

³ OLAP: es el acrónimo en inglés de procesamiento analítico en línea (On-Line Analytical Processing).

- Total Recall

La versión Personal está destinada al uso de desarrolladores es una versión mono licencia, y ofrece los mismos niveles de funcionalidad de que edición empresarial. (Athreya, Junio 2007)

2.2.2. SQL Server 2012

SQL Server 2012 es un Gestor de Base de Datos que permite un desarrollo y proporciona una amplia inteligencia empresarial lo que le permite ser una base sólida sobre la que las empresas de tamaño pequeño mediano y grande pueden construir su infraestructura de IT. En el núcleo de SQL Server 2012, va a poder encontrar funcionalidades como: Servicio de motor de base de dato, Analysis Services, Integration Services, Reporting Services, Service Broker, Master Data Services, Servicios de calidad de datos. (Stanek, 2013)

Servicio de motor de bases de datos: Es el servicio principal para almacenar, procesar y proteger los datos. El Motor de base de datos proporciona acceso controlado y procesamiento de transacciones rápido para cumplir con los requisitos de las aplicaciones consumidoras de datos más exigentes de su empresa. (Rabeler, 2017)

El Motor de base de datos se utiliza para crear bases de datos relacionales para el procesamiento de transacciones en línea o datos de procesamiento analíticos en línea. Se pueden crear tablas para almacenar datos y objetos de base de datos como índices, vistas y procedimientos almacenados para ver, administrar y proteger los datos. Puede usar SQL Server Management Studio para administrar los objetos de bases de datos y SQL Server Profiler para capturar eventos de servidor. (Rabeler, 2017)

Analysis Services: Este componente de SQL Server incluye las herramientas para crear y administrar aplicaciones de procesamiento analítico en línea (OLAP⁴) y de minería de datos. (Rabeler, 2017)

Integration Services: Es un conjunto de herramientas gráficas y objetos programables para mover, copiar y transformar datos. También incluye el componente Data Quality Services (DQS) para Integration Services. (Rabeler, 2017)

Reporting Services: Incluye componentes de servidor y de cliente para crear, administrar e implementar informes tabulares, matriciales, gráficos y de forma libre. Reporting Services es también una plataforma extensible que puede utilizarse para desarrollar aplicaciones de informes. (Rabeler, 2017)

Service Broker: Se encarga de proveer un soporte nativo para los mensajes y las consultas de las aplicaciones en el motor de base de datos de SQL Server. El objetivo es facilitar a los desarrolladores la creación de aplicaciones avanzadas que usan componentes del motor de base de datos para comunicarse entre diferentes bases de datos. Los desarrolladores pueden usar Service Broker para construir de manera sencilla aplicaciones distribuibles y confiables. (Stanek, 2013)

Master Data Services: Es la solución de SQL Server para la administración de datos maestros. MDS se pueden configurar para administrar cualquier dominio (productos, clientes, cuentas) e incluye jerarquías, seguridad específica, transacciones, creación de versiones de datos y reglas de negocios, así como un Complemento para Excel que se puede usar para administrar datos. (Stanek, 2013)

Servicios de calidad de datos: “Proporciona un marco de trabajo para crear un repositorio con una base de conocimiento de metadatos, con el fin de mejorar la calidad de los datos de su organización. Los procesos de limpieza de los datos pueden modificar o eliminar datos incompletos o incorrectos. Asimismo, los procedimientos de

⁴ OLAP: es el acrónimo en inglés de procesamiento analítico en línea (On-Line Analytical Processing).

coincidencia de datos pueden identificar y combinar los duplicados según sea necesario.”
(Stanek, 2013)

2.2.3. Otros DBMS Sql

Adicional a Oracle y SQL Server, en el mercado existen otros DBMS Sql como:

- IBM DB2 que se considera una de las bases de datos más utilizada en entornos Unix/Linux después de Oracle.
- Teradata destaca sobre los demás tipos de bases de datos por su capacidad de almacenamiento y de análisis de datos. Suele ser utilizado en grandes instalaciones de Big Data.
- SAP Sybase aunque ya no es muy utilizada, sigue destacando por su escalabilidad y rendimiento.
- Informix, es parte de la familia de productos RDBMS de IBM, adquirida en 2001 a una compañía (también llamada Informix o Informix Software) cuyos orígenes se remontan a 1980. (Montalvo, 2019)

Existen también DBMS relacionales libres cuyos beneficios son, la posibilidad de revisar e incluso modificar el código a nuestra conveniencia y la no necesidad de pago, como:

- MySQL, sistema de gestión de bases de datos relacional desarrollado bajo licencia dual: Licencia pública general/Licencia comercial por Oracle Corporation y está considerada como la base datos de código abierto más popular del mundo, y una de las más populares en general junto a Oracle y Microsoft SQL Server, sobre todo para entornos de desarrollo web.
- Maria DB, uno de los más populares sistemas de gestión de bases de datos libres, fue desarrollada por los programadores de MySQL lo que garantiza que se mantendrá como software Libre (Open Source), es utilizada por sus alto performance y estabilidad, con amplias opciones de motores de almacenamiento y plugins que hace de este gestor uno de los más usados como por ejemplo en Wikipedia.
- PostgreSQL, sistema de gestión de bases de datos relacional orientado a objetos y libre, publicado bajo la licencia PostgreSQL, similar a la BSD o la MIT. Como

muchos otros proyectos de código abierto, el desarrollo de PostgreSQL no es manejado por una empresa o persona, sino que es dirigido por una comunidad de desarrolladores que trabajan de forma desinteresada, apoyados por organizaciones comerciales. Dicha comunidad es denominada el PGDG (PostgreSQL Global Development Group).

2.3. DBMS No Sql

Se puede decir que la aparición del término NoSQL aparece con la llegada de la web 2.0 ya que hasta ese momento sólo subían contenido a la red aquellas empresas que tenían un portal, pero con la llegada de aplicaciones como Facebook, Twitter o Youtube, cualquier usuario podía subir contenido, provocando así un crecimiento exponencial de los datos.

Es en este momento cuando empiezan a aparecer los primeros problemas de la gestión de toda esa información almacenada en bases de datos relacionales. En un principio, para solucionar estos problemas de accesibilidad, las empresas optaron por utilizar un mayor número de máquinas, pero pronto se dieron cuenta de que esto no solucionaba el problema, además de ser una solución muy cara. La otra solución era la creación de sistemas pensados para un uso específico que con el paso del tiempo han dado lugar a soluciones robustas, apareciendo así el movimiento NoSQL. (AcensWhitepaper, 2014)

A continuación, se menciona dos de los gestores de bases de datos NoSQL más representativos como son MongoDB y AmazonDB.

2.3.1. MongoDB

MongoDB es un sistema de base de datos multiplataforma orientado a documentos, de esquema libre. Esto significa que cada entrada o registro puede tener un esquema de datos diferente, con atributos o “columnas” que no tienen por qué repetirse de un registro a otro. Está escrito en C++, lo que les confiere cierta cercanía a recursos de hardware de la máquina, de modo que es bastante rápido a la hora de ejecutar sus tareas. Además, está

licenciado como GNU⁵ AGPL⁶ 3.0, de modo que se trata de un software de licencia libre. Funciona en sistemas operativos Windows, Linux, OS X y Solaris. (Aula 301, 2018)

Las características que más destacarían de MongoDB son su velocidad y su rico pero sencillo sistema de consulta de los contenidos de la base de datos. Se podría decir que alcanza un balance perfecto entre rendimiento y funcionalidad, incorporando muchos de los tipos de consulta que se puede utilizar en cualquier sistema relacional, pero sin sacrificar en rendimiento. (GenBeta, 2011)

2.3.2. Amazon DynamoDB

Amazon DynamoDB es un servicio de bases de datos NoSQL totalmente administrado que ofrece un desempeño rápido y previsible, así como una escalabilidad óptima. DynamoDB le permite delegar las cargas administrativas que supone tener que utilizar y escalar bases de datos distribuidas, para que no tenga que preocuparse del provisionamiento, la instalación ni la configuración del hardware, ni tampoco de las tareas de replicación, aplicación de parches de software o escalado de clústeres. Además, DynamoDB ofrece el cifrado en reposo, que elimina la carga y la complejidad operativa que conlleva la protección de información confidencial. (Amazon Web Services, 2018)

Amazon DynamoDB proporciona una funcionalidad de backup bajo demanda. Le permite crear backups completos de las tablas para una retención y archivado a largo plazo con el objetivo de cumplir los requisitos de conformidad normativa. DynamoDB permite eliminar automáticamente los elementos vencidos de las tablas, para ayudarle a reducir el consumo de almacenamiento y el costo que suponen los datos que ya no son pertinentes. (Amazon Web Services, 2018)

2.4. Definición de Data

La palabra datos se deriva del vocablo latín *datum* “lo que se da”; por lo tanto, los datos en realidad son hechos dados, a partir de los cuales es posible inferir hechos adicionales. Inferir hechos adicionales a partir de hechos dados es exactamente lo que hace un DBMS

⁵ GNU: es el acrónimo recursivo de "GNU's Not Unix" (en español: GNU no es Unix)

⁶ AGPL La licencia pública general de Affero (en inglés, Affero General Public License, también Affero)

cuando responde a una consulta de un usuario. Un “hecho dado” corresponde a su vez a lo que en lógica se denomina proposición verdadera; por ejemplo, el enunciado “El proveedor X1 se ubica en Quito” podría ser una de estas proposiciones verdaderas. De aquí se desprende que una base de datos es en realidad una colección de tales proposiciones verdaderas. (Date, 2001)

Los datos se representan como pulsaciones o pulsos electrónicos a través de la combinación de circuitos (denominados señal digital), desde el punto de vista de la computación. Pueden ser:

- Datos alfanuméricos (A ... Z, a ... z, 0 ... 9).
- Datos numéricos (Enteros, Decimales, Reales)
- Datos temporales (Fechas, Horas)
- Datos simbólicos o caracteres especiales (por ej. %, \$, #, @, &, etc.)
- Datos multimedia (imágenes, audios, videos)
- Datos espaciales (Latitud, Longitud, Altitud)

Esos datos, cuando se trabaja en una computadora, son convertidos en números dígitos que, a su vez, son representados como pulsaciones o pulsos electrónicos o estados magnéticos o estados ópticos. En la actualidad para comunicarnos, expresarnos y guardar nuestra información, según se trate de valores numéricos o de texto, se utiliza el sistema de numeración decimal y el alfabeto. Una computadora como funciona con electricidad reconoce dos clases de mensajes: cuando hay corriente eléctrica el mensaje es sí y cuando no hay corriente, el mensaje entonces será no. Para representar un valor dentro de una computadora se usa el sistema de numeración binario, que utiliza sólo dos dígitos: el cero (0) y el uno (1). (Tecnología e Infomática, 2009)

2.4.1. **DATA** en una Base de Datos Relacional

La DATA almacenada en un motor de base de datos tiene la abstracción lógica implícita por el tipo de dato definido en los metadatos y estos se podrían describir de la siguiente manera.

2.4.1.1. Datos Compartidos

Cuando se almacenan datos en una base de datos no se suele hacer para el uso exclusivo de una sola persona. Normalmente se espera que una base de datos sea utilizada por más de una persona y podría ser inclusive al mismo tiempo. (Beynon-Davies, 2014)

2.4.1.2. Integración de los datos

Al compartir datos se proporcionan numerosas ventajas a una organización, que solo se materializan si la base de datos se gestiona de una forma responsable. Una de las principales responsabilidades en el uso de una base de datos es asegurar que los datos están integrados. Esto supone que una base de datos debería ser una colección de datos en la cual sería ideal que no existieran datos redundantes. Los datos redundantes son aquellos datos duplicados de forma innecesaria. (Beynon-Davies, 2014)

2.4.1.3. Integridad de los datos

La integridad de un dato alude a ese atributo o cualidad que es inherente a la información cuando se considera exacta, completa, homogénea, sólida y coherente con la intención de los creadores de los datos que la conforman. Esta cualidad, que va ligada al propio dato y no al lugar donde se almacena, al contrario de lo que sostienen creencias bastante extendidas; se obtiene cuando se impide eficazmente que el contenido de una base de datos, de un proceso o de un sistema se vea, accidental o intencionalmente:

- Modificado, en base a su propio contenido o con ayuda de la inserción de nuevo contenido.
- Destruído total o parcialmente. (Power Data, 2013)

2.4.1.4. Seguridad de los datos

Con frecuencia se asocia a la seguridad de los datos con el término de la integridad de estos, pero ambos conceptos son bastante diferentes. La seguridad se refiere a la protección de los datos contra su revelación, su alteración o su destrucción no autorizadas, mientras que la integridad se refiere a la precisión o validez de esos datos. En otras palabras, se puede decir que:

- Seguridad significa proteger los datos ante usuarios no autorizados.

- Integridad significa protegerlos de usuarios autorizados.

La seguridad significa garantizar que los usuarios tengan permiso de hacer las cosas que están tratando de hacer y la integridad involucra asegurar que las cosas que están tratando de hacer sean correctas.

Sin embargo, también existen algunas similitudes: en ambos casos el sistema necesita estar al tanto de ciertas restricciones que los usuarios no deben violar y en ambos casos, estas restricciones deben ser especificadas en un lenguaje adecuado y deben ser mantenidas en el catálogo del sistema; también, en ambos casos el DBMS debe vigilar las operaciones del usuario para asegurarse que cumplan estas restricciones. (Date, 2001)

2.5. Migración de Datos

La migración de datos es el proceso mediante el cual se realiza una transferencia de datos de un sistema de almacenamiento de datos a otro, de unos formatos o tipos de datos a otros o entre diferentes sistemas informáticos (ERP). Habitualmente, un proyecto de migración de datos se lleva a cabo para reemplazar o actualizar servidores o equipos de almacenamiento, para una consolidación de un sitio web, para llevar a cabo el mantenimiento de un servidor o para reubicar un centro de datos. O en algunos casos por una directriz de la Alta Gerencia. (PowerData, 2018)

2.5.1. Tipos de Migración

A continuación, se va a revisar los tipos de migración que se pueden realizar de acuerdo a cada uno de los escenarios que se podría tener desde un punto de vista de cómo se administra la data:

2.5.1.1. Migración de almacenamiento:

Se presenta cuando la problemática está asociada a los límites de capacidad de almacenamiento en donde están alojados los datos. Este tipo de migración no debería presentar mayor problema, siempre y cuando la aplicación sólo utilice interfaces generales para acceder a los datos. En la mayoría de los sistemas esto no es ningún

inconveniente, aunque, cuando se trate de aplicaciones antiguas que se ejecutan en sistemas propietarios, sí podría serlo. (PowerData, 2018)

2.5.1.2. Migración de la base de datos

Se denomina migración de base de datos, al proceso que tiene por objeto tanto la importación como la exportación de una determinada información almacenada en un sistema de bases de datos, para llevar a cabo su traspaso. En este contexto, se trata de exportar los datos a un nuevo sistema con mayor capacidad o más funciones adicionales. Estos cambios llevan consigo una adaptación de todos los datos de una base de datos a otra. Por tanto, siempre que se producen cambios de un sistema de gestión a otro, se habla inevitablemente de los procesos de migración de datos. (Ochando, 2014)

2.5.1.3. Migración de aplicaciones:

Al enfrentarse a una iniciativa de este tipo es imperativo el recurrir a un proceso ETL⁷ completo. Esto es así debido a que, incluso cuando las aplicaciones están diseñadas por el mismo proveedor, almacenan datos en formatos y estructuras significativamente diferentes. Esta particularidad complica la transferencia de datos, y aumenta los tiempos en los que se podría dar este tipo de migración. (PowerData, 2018)

2.5.1.4. Codificación de Data

En este caso, la complejidad radica en el hecho de que la mayoría de los sistemas desarrollados en la plataforma basada en el Computador utilizan codificación ASCII⁸. Sin embargo, los sistemas mainframe se basan principalmente en la codificación EBCDIC⁹ que es incompatible con ASCII y la conversión es necesaria para mostrar los datos. Para poder superar este desafío hay que tener previsto al elegir herramientas ETL que éstas deben soportar las conversiones entre conjuntos de caracteres, incluyendo EBCDIC. (PowerData, 2018)

⁷ ETL: Extract, Transform, Load extraer, transformar y cargar es el proceso que permite a las...

⁸ ASCII: acrónimo inglés de American Standard Code for Information Interchange Código...

⁹ EBCDIC: Extended Binary Coded Decimal Interchange Code, Código de intercambio decimal de código...

2.6. Definición de ERP

Los sistemas de planificación de recursos empresariales son sistemas de gestión de información que automatizan muchas de las prácticas de negocio asociadas con los aspectos operativos o productivos de una empresa. Básicamente es una arquitectura de software para empresas que facilita e integra la información entre las funciones de manufactura, logística, finanzas y recursos humanos de una empresa. En la mayoría de los negocios online, esto hace referencia a todo el tema de la facturación, gestión de pedidos y envíos, así como datos de los clientes, del personal, distribuidores y otros. Actualmente, un ERP se ha convertido en un aspecto fundamental en la elaboración y gestión de una estrategia empresarial. (Conceição Menezes, 2010)

2.6.1. Ventajas de un ERP

De manera general, y con una visión en los procesos que tiene una organización algunas de las ventajas que se pueden tener del uso de un ERP son:

- Optimización de los procesos empresariales.
- Acceso a la información.
- Posibilidad de compartir información entre todos los componentes de la organización.
- Integración de las distintas bases de datos de una compañía en un solo programa.
- Ahorro de tiempo y costes.
- Eliminación de datos y operaciones innecesarias de reingeniería.

Por otro lado, los ERP ofrecen integración con soluciones de Inteligencia de Negocios, los mismos que permiten realizar informes sobre el estado de la empresa directamente con los datos del sistema ERP. Esto ofrece un nivel de conocimiento detallado y actualizado de la situación de la empresa que resulta indispensable para la toma de decisión y también como insumo para mejorar procesos internos como las ventas, finanzas, la organización u otros aspectos clave de una compañía. (Grupo CONSISA, 2017)

2.6.2. Desventajas de un sistema ERP

El inconveniente más común suele ser el coste del software ERP. Esto se debe normalmente al nivel de personalización que necesita un sistema ERP para cubrir las necesidades de la empresa: a mayor nivel de personalización, mayor precio. Además, algunos de los costes que un ERP conlleva pueden aparecer de forma posterior a su instalación y adquisición, lo que se denomina costes ocultos.

Otra desventaja a tener cuenta al elegir un paquete ERP está en la implementación. En algunos casos, puede que la instalación, el hardware necesario para su funcionamiento y la preparación de su infraestructura tarden demasiado. Esto puede provocar retrasos en el funcionamiento interno de su empresa que pueden causar pérdidas. No obstante, existen sistemas en la nube que pueden prevenir este tipo de inconvenientes, al no tener que ser implementados físicamente en la empresa. (tic.portal, 2018)

3. CAPÍTULO II: Situación Actual de Proauto C.A.

Proauto C.A. empresa dedicada a la venta y postventa de vehículos de la marca Chevrolet en la modalidad de concesionario; realizó entre octubre del 2016 y marzo del 2017 un cambio de ERP entre un sistema desarrollado con Visual Basic 6.0 con SQL Server a un sistema desarrollado con Designer de Oracle con base de datos Oracle 11g. Esto generó una migración de los datos entre los dos motores de base de datos.

A continuación, se revisará información referente a la empresa, que servirá para el presente trabajo de titulación.

3.1. Historia

Proauto C.A. es un concesionario de vehículos de la marca Chevrolet, fundada hace más de 20 años, empieza sus operaciones como un taller de reparación vehicular y venta de vehículos usados; posteriormente obtiene la representación de Colmotores para vender vehículos pesados de más de 4 toneladas y tractomulas en el Ecuador.

En 1992 General Motors entrega la distribución de todo tipo de vehículos livianos fabricado por GM, en ese entonces las ventas ascendían a 26 unidades al mes.

Hoy en día es el concesionario con más market share de vehículos Chevrolet en el Ecuador, aproximadamente un 30% y cuenta con más de 200 colaboradores y cinco sucursales, cuatro en Quito y una en Cayambe. El nicho de mercado que cubre Proauto son vehículos livianos y camiones, gran parte de los automotores son producidos en la planta de Omnibus BB y hay ediciones especiales que se importan directamente de Korea, Venezuela entre otros países. (Proauto C.A., 2018)

En la actualidad PROAUTO C.A. es uno de los concesionarios más representativos dentro de la red Chevrolet, por ser parte de un grupo corporativo de concesionarios Chevrolet en las principales ciudades del país como son Quito, Cuenca y Guayaquil, lo que le permite tener ventas mensuales en promedio de 200 unidades, e ingresos a sus talleres de 1800 órdenes de trabajo al mes, distribuidas entre sus 5 agencias.

3.2. Líneas de Negocio

Dentro de las líneas de Negocio más importantes de Proauto C.A., están la comercialización de vehículos como la fuente principal de ingresos de la compañía, pero también es importante mencionar, que la empresa cuenta con varios talleres en la ciudad de Quito, además de un amplio inventario de repuestos de la línea Chevrolet.

A continuación, se revisará de manera macro el proceso que se maneja en cada una de las líneas de negocio más importantes.

3.2.1. Comercialización de Vehículos

Actualmente las ventas se encuentran en un promedio de 200 vehículos livianos y entre 20 pesados, los modelos que más se comercializan son los de la línea de pasajeros que incluyen Aveo Family, Spark y Sail, sin embargo, también por el manejo de diversas flotas que tiene el concesionario se comercializa un número importante de unidades de camionetas.

Para poder conseguir este número de ventas el concesionario cuenta con procesos definidos y apegados a ciertos estándares de la marca, a más de realizar una importante inversión en tecnología, lo que permite que la facturación de estas unidades se realice de una manera óptima. Adicionalmente PROAUTO C.A. cuenta con un ERP que automatiza y controla los diferentes procesos.

El proceso de comercialización de vehículos inicia desde la prospección, esto genera tráfico en el concesionario. Los clientes que llegan son recibidos por una anfitriona la que se encarga de registrar en la Base de Datos, la información correspondiente a números de cédula, nombres, dirección, teléfonos, y medio u origen de contacto. A continuación, el cliente es asignado a un asesor de ventas el que se encarga de generar una cotización, en

la que se incluirán los datos de la negociación como por ejemplo: datos del modelo de vehículo, las condiciones de pago, con o sin accesorios, descuentos y por último se genera un documento con toda la información.

3.2.2. Servicio de Talleres

Uno de los negocios más importantes de PROAUTO C.A. es la Postventa. PROAUTO C.A. maneja actualmente 7 talleres que están distribuidos de la siguiente manera:

El taller de Matriz, que tiene una capacidad para recibir 780 vehículos al mes aproximadamente. Puede gestionar mantenimientos preventivos y correctivos, además de garantías.

El taller de Monteserrín, ubicado en la agencia con el mismo nombre tiene la capacidad instalada para atender 200 vehículos. De igual manera, este taller puede realizar trabajos preventivos y correctivos, con la particularidad que aquí se trabajan especialmente Vehículos de Flotas; lo que hace que este taller sea además multimarca.

El taller de Carapungo, la agencia más grande de PROAUTO C.A.; abarca 3 talleres en su localidad, un taller de vehículos livianos, uno para vehículos pesados y un taller que maneja el servicio de colisiones. Cada uno con una capacidad de atención de 460 para livianos, 150 para pesados y 122 vehículos en colisiones.

Cada taller cuenta con técnicos especializados en servicio de la marca Chevrolet. El proceso que maneja el taller para la recepción de sus vehículos está estandarizado por la marca con el proceso de Servicio Personalizado Chevrolet, el cual determina que todo vehículo debe ingresar al taller con una cita, la misma que debe ser registrada en una agenda virtual distribuida por bahías, cada bahía recibe en promedio 10 Vehículos en el día y su capacidad está determinada por el número de horas que cada mantenimiento tiene, de esta manera se agendan en un periodo disponible de 8 horas.

Una vez que el cliente tiene una cita asignada, al llegar al concesionario es recibido directamente en la bahía de trabajo por el Técnico, el mismo que apoyado en el ERP tiene la información del cliente, el vehículo y el mantenimiento a realizar lo que ayuda a la optimización de tiempos en la recepción del vehículo.

3.2.3. Venta de Repuestos

La venta de repuestos es otro de los procesos importantes de Proauto y representan un 10% del ingreso total de la compañía. Dentro de esta línea de negocios existen dos canales, el primero es la venta de repuestos por mostrador, y el segundo es la venta de repuestos por los ingresos al taller.

Dentro de la venta de repuestos por mostrador de igual manera se manejan algunos canales, como es la venta al público en general, venta a aseguradoras y venta a clientes que son parte de una flota. El sistema que maneja la compañía está encargado de administrar todo el inventario de la compañía y brindar las herramientas necesarias para dar el soporte a la gestión de repuestos en todos los canales mencionados.

Por otro lado, la administración del inventario en los talleres esta de igual manera soportado por el sistema de la compañía la diferencia es que aquí el manejo del inventario o el descargo del inventario está vinculado con las ordenes de trabajo que se abren en el taller, este inventario pasa al estado “en proceso”, mientras la orden de taller no sea facturada.

3.3. Organigrama de la Empresa

La estructura organizacional de PROAUTO C.A. está conformada de la siguiente manera, Gerente General, Gerencia por Línea de Negocio, es decir Gerente Comercial, Gerente de Post Venta, Gerente de Repuestos, Gerente de Pesados, y además por las respectivas Gerencias Administrativas como son Gerencia Financiera, Gerencia de Talento Humano, y Jefaturas que reportan directamente a la Gerencia General como es el caso de la Jefatura de Sistemas, Jefatura de Marketing y Jefatura de GM Difference o Calidad.

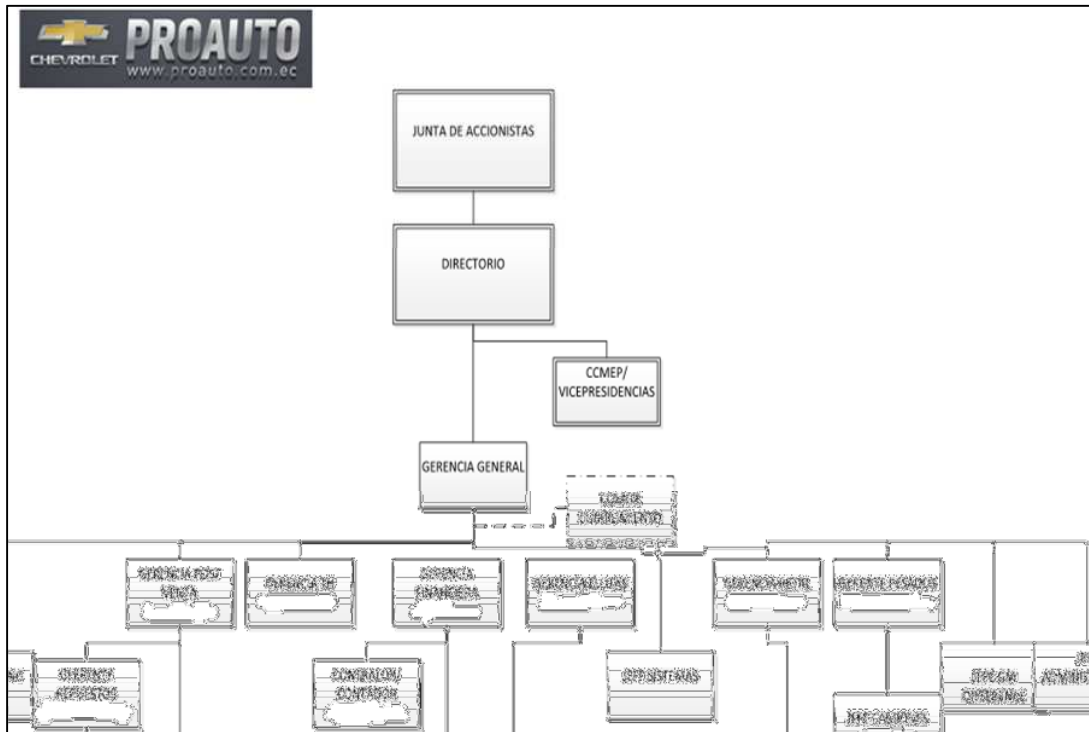


Figura 2-1 Organigrama General de Proauto. (Montalvo, 2019)

Como se puede observar en la Figura 2-1 la Jefatura del área de Sistemas está en un nivel inferior a las Gerencias de línea, sin embargo, por ser esta un área de apoyo estratégico dentro de la organización tiene reporte directo con la Gerencia General, esto permite que los proyectos de tecnología sean gestionados con mayor agilidad para más información referirse al Anexo 1.

3.4. Situación Financiera

Para el presente trabajo de titulación se tomará la información financiera de la empresa con corte al mes de octubre del 2018, revisando principalmente las líneas más importantes de la compañía como son venta de vehículos y servicio de talleres.

Dealer	Objetivo total				CUMPLIMIENTO			
	Retail	Flotas	Chevyplan	Total	Retail	Flotas	Chevyplan	TOTAL
Autolandia S.A.	114	18	18	150	102%	89%	83%	98%
Automotores Continental S.A. Quito	244	70	16	330	98%	41%	88%	85%
Ecua-Auto S.A.	103	22	20	145	114%	177%	95%	121%
Latinoamericana De Vehiculos C.A.	123	33	0	156	83%	161%		99%
Metrocar S.A. - Quito	161	31	9	201	88%	132%	78%	94%
Proauto C.A.	164	54	16	234	104%	152%	106%	115%
Vallejo Araujo S.A.- Quito	139	37	6	182	114%	111%	33%	110%
Total Pichincha	1048	265	85	1398	99,4%	113,6%	87,1%	101,4%

Figura 2-2 Cierre de Ventas Meta GM (Ecuador, 2018)

La Figura 2-2 representa las **ventas de vehículos** correspondientes al mes de **octubre del 2018** correspondiente a los Dealer de la zona 1 (Zona Quito). Este cuadro registra el número de vehículos en unidades de acuerdo con las ventas de Retail, Flotas y Chevyplan. Seguido a la derecha por las columnas que representan el porcentaje de cumplimiento de cada Dealer, en donde Proauto C.A. tiene un cumplimiento del 115% sobre el objetivo establecido por la marca con ventas totales de 234 vehículos en octubre 2018.

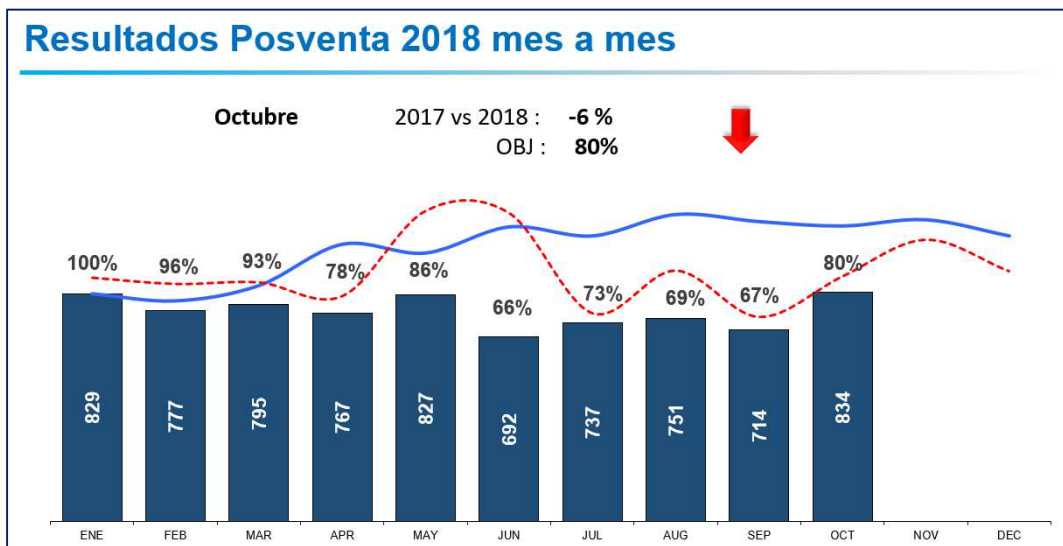


Figura 2-3 Cierre Ventas Post Venta 2018

La figura 2-3 representa los **resultados de posventa del año 2018 mes a mes**, se puede apreciar el evolutivo de ventas de Pos Venta incluyendo mano de obra, repuestos y accesorios para el año 2018. La línea azul indica el presupuesto de la compañía para el área de Post Venta, mientras que la línea roja representa las ventas del año anterior. En donde en el mes de octubre se observa un cumplimiento con el presupuesto del 80%. Las barras azules corresponden a las ventas y están representadas en miles de dólares. Están

incluidas ordenes de trabajo de clientes Retail, Flotas y servicio de Colisiones, es decir en octubre Proauto C.A. facturo \$834.000 en el área de postventa.

3.5. Procesos del Negocio

A continuación, se va a revisar de manera macro los procesos más importantes y en los cuales se realizará el análisis de la data a ser migrada, basado en las líneas de negocios del área comercial, el área de Pos Venta y el área de repuestos como se mencionó en este capítulo.

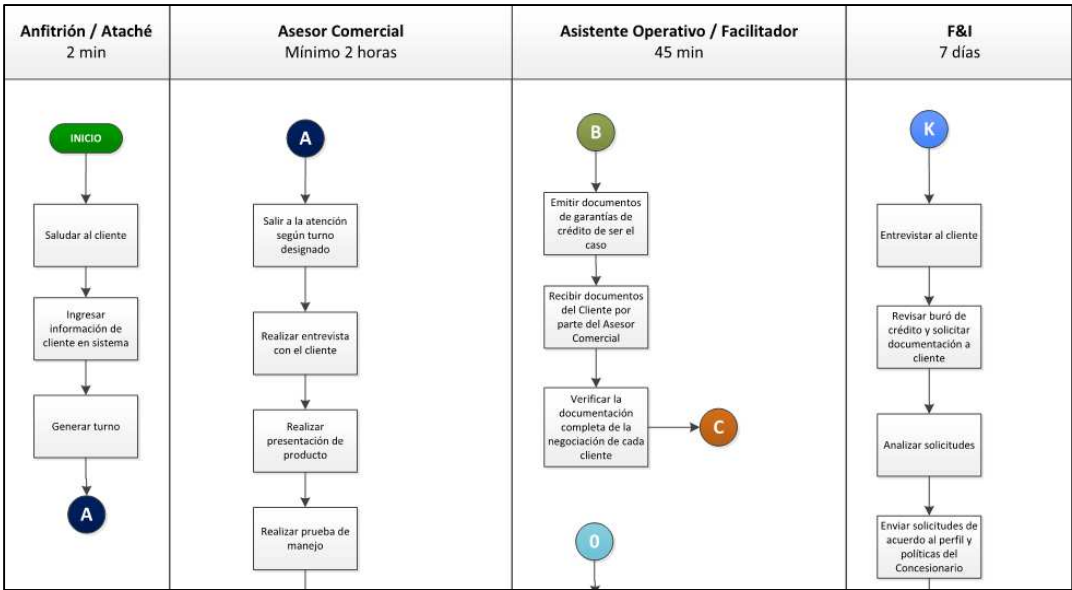


Figura 2-4 Diagrama Proceso Área Comercial

La Figura 2-4 representa un extracto del proceso del área comercial que inicia en el anfitrión, quien recibe al cliente y es el encargado de ingresar los datos del cliente en el sistema y generar un turno. Este turno es enviado al asesor comercial, el mismo que realiza la atención al cliente, genera una cotización y realiza varios pasos incluidos dentro del proceso. Si se cierra la venta se involucran otros actores como son: el asistente operativo y la F&I. Para ver el diagrama completo referirse al Anexo 2.

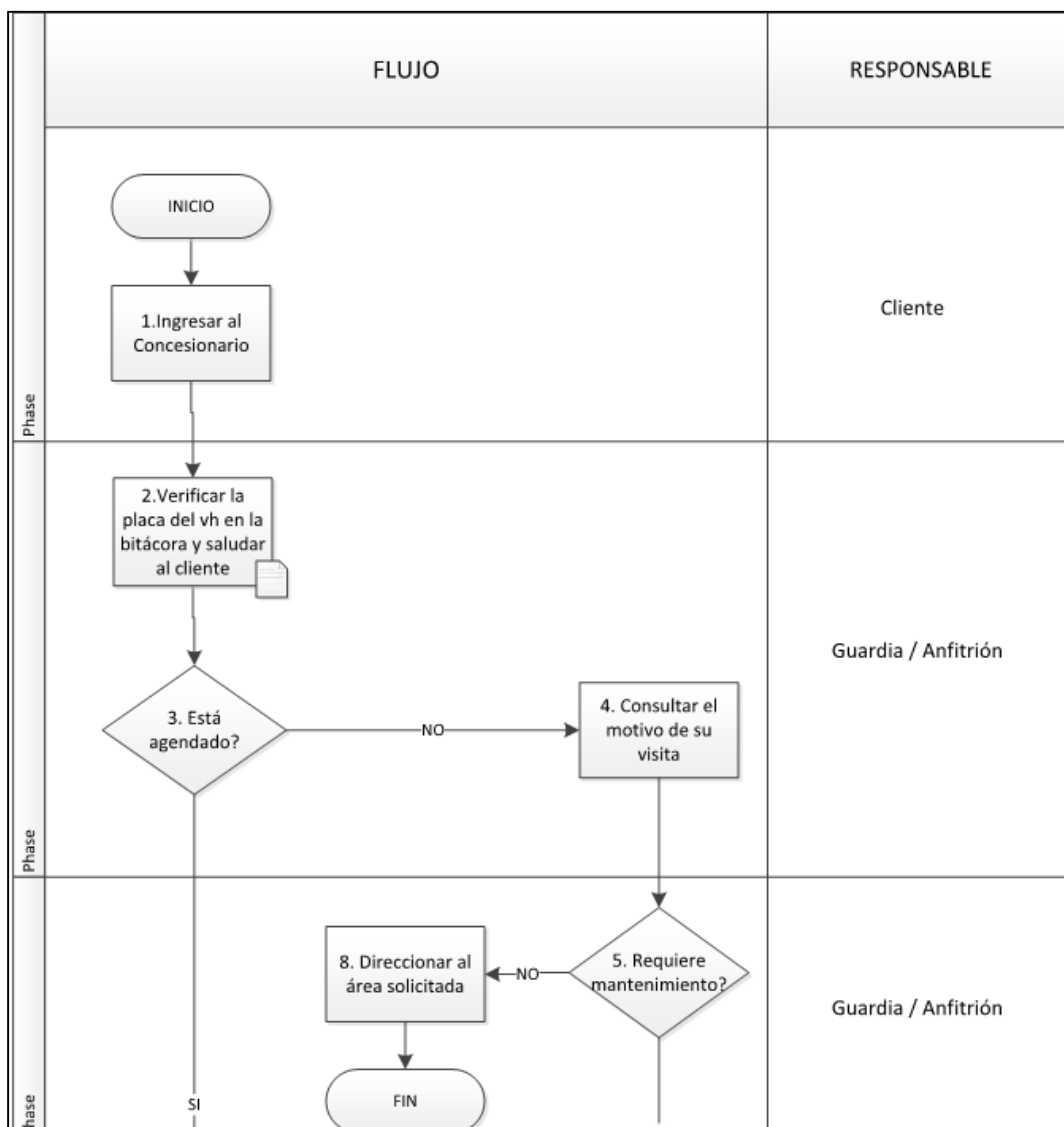


Figura 2-5 Diagrama Proceso Talleres

La Figura 2-5 corresponde al diagrama del proceso de recepción de vehículos en el taller. El proceso empieza en la recepción del cliente por el guardia, el mismo que con la placa del vehículo tiene en el sistema la información del motivo de ingreso del cliente y sus datos. Si el cliente tiene agendada una cita, el guardia se encarga de direccionar al cliente a una bahía de servicio, caso contrario el cliente debe acercarse al balcón de servicio en donde un asesor de servicio le generará una cita en el sistema para posteriormente generar una orden de trabajo y poder realizar el mantenimiento al vehículo. Para mayor detalle referirse al Anexo 3.

4. CAPÍTULO III: ERP DMS

En este capítulo se revisarán las características generales del sistema DMS, cuando fue implementado en PROAUTO C.A., cómo ha sido su uso en la compañía y además sobre que infraestructura trabaja. También se revisará el modelo entidad relación de las principales tablas y qué data es la que se almacena. Esto será analizado en el proceso de migración.

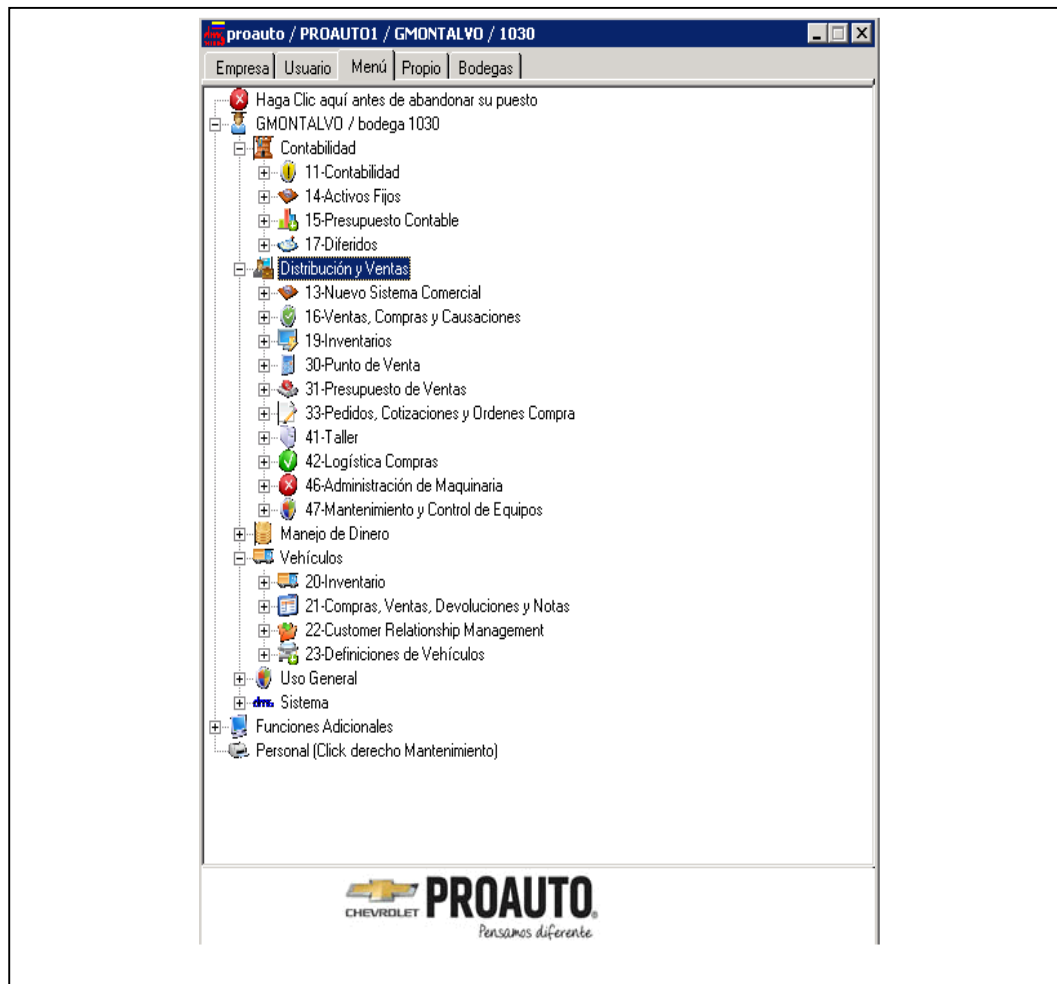


Figura 3-1 Pantalla principal DMS (Montalvo, 2019)

La Figura 3-1 representa la pantalla de acceso principal del ERP¹⁰ DMS, desde esta

¹⁰ ERP: sistemas de planificación de recursos empresariales en inglés, enterprise resource planning

pantalla el usuario puede acceder a los diferentes módulos de sistema, como se puede ver es un sistema completamente modular, y de manera rápida se puede observar los módulos de las principales líneas de negocios como son Ventas, Talleres y manejo de inventarios.

4.1. Generalidades

El ERP DMS (Dynamic Modular System), fue implementado en PROAUTO C.A. en el mes de junio del 2006 por la empresa DMS Ecuador, quienes en conjunto con DMS Colombia (desarrolladores) realizaron la implementación. Este sistema está desarrollado en Visual Studio 6.0, es un sistema que funciona bajo la arquitectura cliente servidor. Además, consta de todas las funcionalidades que un concesionario de vehículos necesita, es decir módulos para el manejo de inventarios de vehículos y repuestos, un módulo para el proceso comercial, así como también módulos para el manejo de taller y además toda la administración contable de la compañía.

Por otro lado, el sistema tiene integración con Crystal Reports para la generación de informes gerenciales y tiene la ventaja que se pueden personalizar estos reportes de acuerdo a la necesidad de cada cliente de manera autónoma, es decir sin contar con el soporte de la empresa desarrolladora del sistema.

El licenciamiento de DMS es por usuario concurrente, sin embargo, por ser PROAUTO C.A. uno de los primeros concesionarios en Ecuador que empezaron a trabajar con este ERP, se negoció un licenciamiento ilimitado de usuarios, esta ha sido una ventaja también en el uso de la herramienta que ha permitido la estabilidad en su uso durante más de 10 años.

El Sistema gestor de base de datos que soporta este ERP es SQL Server, con un total de 950 tablas, 603 vistas, 524 Procedimientos Almacenados, y 38 funciones que almacenan y gestionan la transaccionalidad y reportería del sistema DMS. En la actualidad la base de datos tiene una capacidad ocupada en disco de 63.5 Gigas correspondiente a datos almacenados desde junio del 2006 hasta el 13 de marzo del 2018.

4.2. Infraestructura

PROAUTO C.A. es una empresa que a través de los años se ha preocupado por ir actualizando su infraestructura tecnológica, debido al incremento en el número de usuarios y la continua optimización y mejora de procesos que Chevrolet y el mercado exigen.

DMS inicio su operatividad en un servidor HP Proliant BL 460c G6, con dos discos de 146Gb SAS de 7.5K, 12 GB de memoria RAM y un solo procesador Intel(R) Xeon(R) CPU E5520 @ 2.27GHz (4 Cores), este servidor es parte de un BLADE HP BladeSystem c3000 Enclosure, el mismo que mantiene redundancia de fuentes de poder y ventiladores. En este equipo se tenía instalado como sistema operativo Windows Server 2003 y el motor de base de datos SQL Server 2005.

Con el tiempo el equipo se quedó corto de espacio en almacenamiento en disco, así como de capacidad de procesamiento de memoria RAM. Por esta razón se presentó a la Gerencia General la necesidad de aumentar discos y memoria RAM ya que el ERP DMS, empezó a tener demoras y generar molestias en los usuarios. Es por esto que se aumentó la memoria RAM a 24 GB y se decidió de igual manera migrar la versión de sistema operativo a Windows server 2008 y SQL server 2008, cada uno con su respectiva licencia y ambos ya trabajando en tecnología de 64 bits.

La instalación de sistema operativo y SQL Server estaba directo al equipo, es decir sin utilizar un sistema de virtualización. El ERP al ser un sistema cliente servidor, de igual manera estaba instalado en este servidor, es decir, este equipo hacia las funciones de servidor de aplicaciones y servidor de base de datos.

Estos cambios o mejoras en el servidor no duraron mucho tiempo y nuevamente se requirió hacer una actualización de Hardware y Software. Es aquí cuando se decide comprar un nuevo servidor HP ProLiant BL460c Gen8.

Este es parte del grupo de servidores instalados en el Blade, este nuevo equipo tiene un procesador de Intel(R) Xeon(R) CPU E5-2640 0 @ 2.50GHz (6 Cores) con 32GB de memoria RAM y 600GB en dos discos SAS de 10K bajo Raid 1. Sobre este equipo se decide colocar Windows Server 2012 y SQL server 2012. Con este cambio se mejoró el

rendimiento del ERP, al igual que el servidor anterior la instalación de SQL Server, Sistema Operativo y ERP fueron hechas directamente al servidor físico, sin la utilización de un software de virtualización. Este equipo estuvo en producción durante un periodo de 3 años desde 2013 hasta finales del 2016.

Durante el siguiente año se solicita nuevamente a la gerencia general la actualización del servidor debido a que el BLADE en el que estaba instalado había perdido ya su garantía con HP y existía un alto riesgo de poder mantener la continuidad del negocio en el caso de que alguna de las partes físicas del Blade fallase. Por esto se decide la compra de un nuevo servidor pero que ya no sea parte del Blade sino un servidor tipo torre, HP ProLiant ML350 Gen9. Este equipo contaba ya con dos procesadores, Intel(R) Xeon(R) CPU E5-2650 v3 @ 2.30GHz de 10/10 cores; 20 threads cada uno, y 64 GB de memoria RAM distribuida en 32 GB para cada procesador.

Además de tener 4 discos duros dos de 450GB SAS de 15K y dos de 600GB de 10K cada uno en RAID 1 respectivamente aquí se decidió utilizar los discos de 450GB para instalar el sistema operativo Windows Server 2012 y el gestor de base de datos SQL Server 2012, y en los otros dos discos el ERP, el archivo de la base de datos estaba también en el disco de 450 por ser de mayor velocidad que los discos de 600GB.

Estos servidores forman parte del Data Center de PROAUTO C.A., el mismo que está diseñado para administrar toda la Red interna de PROAUTO C.A. en su agencia Matriz, además de su central telefónica y todo el cableado estructurado, cuenta con dos UPS de 3KVA que protegen servidores y equipos de comunicación. Este data center cuenta además con aire acondicionado y está en un cuarto debidamente aislado. El acceso a esta área esta únicamente autorizado para el personal del departamento de sistemas. Todo lo antes mencionado permite asegurar la continuidad del negocio.

4.3. DBMS con el que Trabaja

Como ya se mencionó en este capítulo, el ERP DMS es desarrollado en Visual Studio 6.0 y trabaja con una conexión vía ODBC contra SQL Server. En PROAUTO C.A. este sistema empezó funcionando con la versión SQL Server Standard Edition de 32 bits, luego se migro a la versión de SQL Server 2008 Standad Edition R2 ya de 64 bits. En los últimos 4 años se ha venido trabajando con la versión SQL Server 2012 de igual manera

de 64bits. Esta versión tiene una limitación en su licencia para trabajar con 14 Cores y una memoria RAM máxima de hasta 64GB. No incluye soporte para características avanzadas de Alta disponibilidad o características de Inteligencia de Negocios más potentes como PowerPivot y servicios de datos maestros, entre otros. Su licenciamiento se realiza por núcleo o por servidor.

4.3.1. Modelo Entidad Relación en SQL Server

Como se mencionó anteriormente, la Base de Datos del ERP DMS de PROAUTO posee alrededor de 950 Tablas. Entre estas tablas se encuentran las que gestionan la parte operativa de la empresa, es decir las que manejan los procesos más importantes que ya se han analizado anteriormente. Se empezará con el análisis de una sección del modelo entidad relación correspondiente al área de venta de Vehículos.

Este modelo sintetiza el proceso de creación de una cotización de un vehículo con sus respectivas opciones de financiamiento e incluye datos del cliente y del vehículo que se va a cotizar, así como también datos del asesor de ventas y la agencia en la que se genera esta cotización.

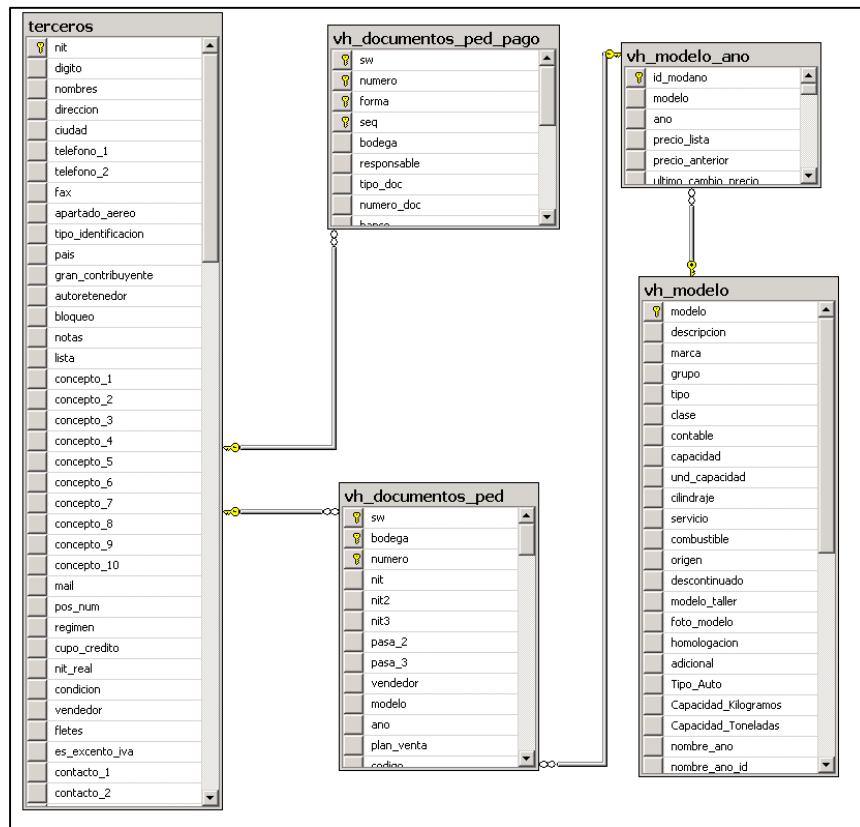
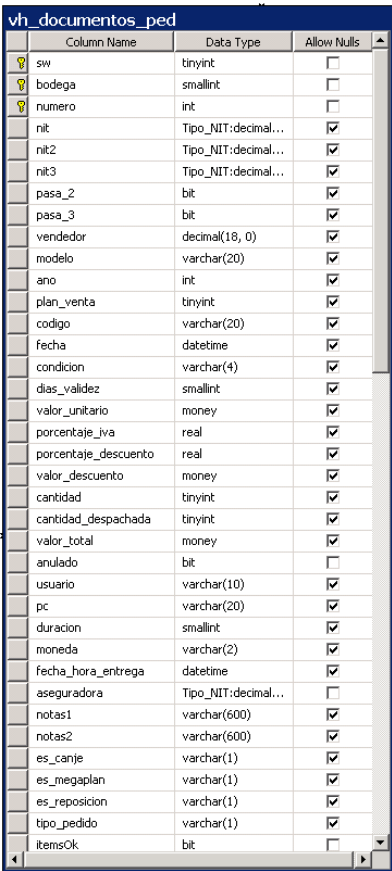


Figura 3-2 Modelo Entidad Relación Cotización de Vehículos (Montalvo, 2019)

La figura 3-2 contiene la relación entre algunas de las tablas que se manejan en DMS. Una de las más importantes es la tabla de terceros, en la que se almacenan todos los datos de las personas que tienen algún tipo de relación con la compañía como, por ejemplo: clientes, proveedores, empleados, etc. Más adelante se revisará a profundidad el contenido de esta tabla y la relaciones que maneja dentro de todo el ERP; con las distintas tablas que se utilizan en todos los procesos operativos y contables de la compañía.

Al ser esta tabla el maestro de personas, las tablas `vh_documentos_ped` y `vh_documentos_ped_pago`, son las tablas que almacenan los datos de una cotización de un vehículo, es por esta razón que necesitan tener una relación con las tablas `vh_modelo` y `vh_modelo_ano`, las mismas que tienen la información y características de los distintos modelos que ofrece la marca Chevrolet.



Column Name	Data Type	Allow Nulls
sw	tinyint	☐
bodega	smallint	☐
numero	int	☐
nit	Tipo_NIT:decimal...	☒
nit2	Tipo_NIT:decimal...	☒
nit3	Tipo_NIT:decimal...	☒
pasa_2	bit	☒
pasa_3	bit	☒
vendedor	decimal(18, 0)	☒
modelo	varchar(20)	☒
ano	int	☒
plan_venta	tinyint	☒
codigo	varchar(20)	☒
fecha	datetime	☒
condicion	varchar(4)	☒
dias_validez	smallint	☒
valor_unitario	money	☒
porcentaje_iva	real	☒
porcentaje_descuento	real	☒
valor_descuento	money	☒
cantidad	tinyint	☒
cantidad_despachada	tinyint	☒
valor_total	money	☒
anulado	bit	☐
usuario	varchar(10)	☒
pc	varchar(20)	☒
duracion	smallint	☒
moneda	varchar(2)	☒
fecha_hora_entrega	datetime	☒
aseguradora	Tipo_NIT:decimal...	☐
notas1	varchar(600)	☒
notas2	varchar(600)	☒
es_canje	varchar(1)	☒
es_megaplan	varchar(1)	☒
es_reposicion	varchar(1)	☒
tipo_pedido	varchar(1)	☒
ItemsOk	bit	☐

Figura 3-3 Tabla `vh_documento_ped` (Montalvo, 2019)

La Figura 3-3 indica más a detalle la tabla `vh_documento_ped` y se puede ver que los tres primeros campos son el primary key o la clave principal de esta tabla. El campo `sw` de tipo `tinyint` es propio de manejo del ERP. Este campo determina la naturaleza del

documento es decir si es una salida o un ingreso. El campo bodega de igual manera hace referencia a una parametrización que maneja DMS en donde se determina la agencia en donde se está generando la cotización para este caso y el campo número de tipo int identifica el número de documento o cotización. Además, se puede observar los otros tipos de datos que esta tabla maneja como son varchar para los campos de texto, money para campos de manejo de valores de dinero, y datetime para fechas; por otro lado, está el campo modelo y el campo nit que permiten la relación con las tablas vh_modelo_ano y terceros respectivamente. Esta tabla tiene 45025 registros.

terceros			
	Column Name	Data Type	Allow Nulls
	nit	Tipo_NIT:decimal...	☐
	digito	tinyint	☒
	nombres	varchar(250)	☐
	direccion	varchar(150)	☒
	ciudad	varchar(20)	☒
	telefono_1	varchar(15)	☒
	telefono_2	varchar(15)	☒
	fax	varchar(15)	☒
	apartado_aereo	varchar(10)	☒
	tipo_identificacion	char(1)	☒
	pais	varchar(20)	☒
	gran_contribuyente	bit	☐
	autoretenedor	bit	☐
	bloqueo	tinyint	☒
	notas	varchar(250)	☒
	lista	tinyint	☒
	concepto_1	varchar(5)	☒
	concepto_2	varchar(5)	☒
	concepto_3	varchar(5)	☒
	concepto_4	varchar(5)	☒
	concepto_5	varchar(5)	☒
	concepto_6	varchar(5)	☒
	concepto_7	varchar(5)	☒
	concepto_8	varchar(5)	☒
	concepto_9	varchar(5)	☒
	concepto_10	varchar(5)	☒
	mail	varchar(50)	☒
	pos_num	tinyint	☐
	regimen	char(1)	☒
	cupo_credito	money	☒
	nit_real	Tipo_NIT:decimal...	☐

	descuento_fijo2	real	☒
	ruc_ecuador		☒
	clasificado	char(1)	☒
	descuento_fijo_prov	real	☒
	solo_contado_compra	varchar(1)	☒
	Tipo_Natural	char(1)	☒
	es_artesano	char(1)	☒
	fecha_cumple_ter	datetime	☒
	id	int	☐
	creado_gold	bit	☒
	celular	varchar(15)	☒
	tipo_devolucion	varchar(4)	☒
	Usuario	varchar(20)	☒
	es_excento_iva_compra	char(1)	☒
	codigo_alterno	varchar(20)	☒
	es_RISE	char(1)	☒
	y_pais	varchar(5)	☒
	ext_1	varchar(5)	☒
	tipo_direccion	smallint	☒
	Separa_Iva	bit	☒
	Separa_Negocio	bit	☒
	Solo_Remision	bit	☒
	Devolver_Prod	bit	☒
	cupo_credito_vh	money	☒
	otro_descuento	real	☒
	no_regulados	char(1)	☒
	nit_rep	Tipo_NIT:decimal...	☒
	estrato	char(1)	☒
	paginaweb	varchar(80)	☒
	area_labora	varchar(50)	☒
	cargo	varchar(50)	☒
	ext2	varchar(20)	☒
	celular2	varchar(20)	☒
	email2	varchar(50)	☒
	sincronizado	char(1)	☐

Figura 3-4 Tabla terceros (Montalvo, 2019)

La figura 3-4 representa la tabla terceros con 84252 registros, cuya calve principal o primary key es el campo nit, este campo quiere decir “número de identificación tributaria” este es un concepto que se maneja en Colombia, al ser este un sistema desarrollado en ese país se han realizado algunas tropicalizaciones para el manejo tributario de Ecuador.

Es por esta razón que esta tabla también maneja varios campos que son utilizados para este fin como por ejemplo el campo ruc_ecuador que es un campo calculado, además de tener todos los datos de contractibilidad como nombre dirección teléfono, etc., esta tabla

permite realizar cierta personalización en la clasificación de la persona o tercero, como se denomina en DMS, utilizando los campos concepto_1 al concepto_10 estos tienen un relación de uno a muchos con la tablas tercero_1 al tercero_10, la estructura de esta tablas es bastante simple, consta de 3 campos: su clave principal un varchar de 5 que se llama concepto_n, un campo descripción varchar de 40 y un campo exportado que es un char de 1 que funciona como una bandera.

La ventaja es que estas tablas pueden ser llenadas desde una interfaz del ERP con cualquier información que la empresa necesite para clasificar de manera fácil a clientes, proveedores, empresas, etc.

A continuación, se revisará el modelo entidad relación con el que se manejan de manera general la apertura de Ordenes de Trabajo en el taller y todas las tablas que están involucradas en este proceso.

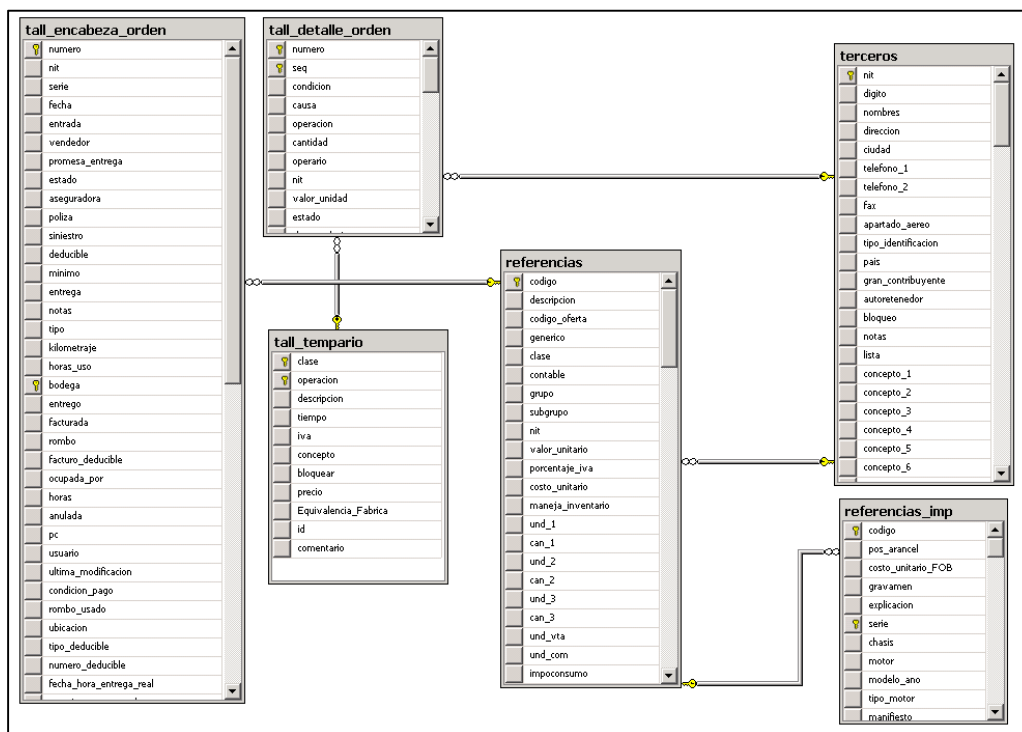


Figura 3-5 Modelo Entidad Relación Apertura de Ordenes en Taller (Montalvo, 2019)

En la Figura 3-5 se puede revisar como es la estructura de las tablas que se encargan de administrar el taller, este es un esquema básico en el que no se pueden ver o analizar las tablas que están involucradas por ejemplo en el proceso de garantías de taller, ni la administración de flotas del taller. Sin embargo, analizando la imagen se puede ver que se maneja una tabla de datos de encabezado que es tal_encabeza_orden, esta tabla maneja

como clave principal los campos número y bodega. El primero es el número de la orden y el segundo es el similar a lo ya revisado en el área comercial, este campo determina la agencia donde se va a crear la orden en esta tabla se almacena el nit del cliente, el código del vehículo el kilometraje con el que ingresa la fecha de ingreso el asesor que crea la orden, entre otros datos.

A continuación, está la tabla tal_detalle_orden que cuenta con la información de detalle de la orden, como son las operaciones de mano de obra, los repuestos, los valores en tiempo y en dólares de cada uno de estos y el técnico asignado. Seguido esta la tabla tal_tempario donde se almacena la información de las operaciones de mano de obra, y la tabla referencias que es otra de las más importantes dentro del sistema; debido a que aquí se almacenan todos los datos de los ítems de inventario, tanto repuestos como vehículos, esta tabla también será analizada más adelante con mayor detalle. Por último, la tabla referencias_imp permite almacenar datos más específicos de vehículos, y la tabla terceros que ya fue analizada.

referencias			
	Column Name	Data Type	Allow Nulls
?	codigo	varchar(20)	☐
	descripcion	varchar(80)	☒
	codigo_oferta	varchar(1)	☒
	generico	varchar(10)	☐
	clase	varchar(10)	☐
	contable	tinyint	☐
	grupo	varchar(10)	☐
	subgrupo	varchar(10)	☐
	nit	Tipo_NIT:decimal...	☐
	valor_unitario	money	☒
	porcentaje_iva	real	☒
	costo_unitario	money	☒
	maneja_inventario	bit	☐
	und_1	varchar(4)	☒
	can_1	real	☒
	und_2	varchar(4)	☒
	can_2	real	☒
	und_3	varchar(4)	☒
	can_3	real	☒
	und_vta	tinyint	☒
	und_com	tinyint	☒
	impoconsumo	real	☒
	valor_und1	money	☒
	valor_und2	money	☒
	valor_und3	money	☒
	conversion	real	☒
	otro_impuesto	real	☒
	minimo_iva	real	☒
	minimo_iva_c	money	☒
	precio_marcado	varchar(1)	☒
	factor_compra	real	☒
	factor_venta_1	real	☒
	factor_venta_2	real	☒
	factor_venta_3	real	☒
	factor_venta_4	real	☒
	factor_venta_5	real	☒
	factor_venta_6	real	☒
	factor_venta_7	real	☒

	costo_compra_emerg...	money	☒
	codigo_descuento	char(1)	☒
	pedir	char(1)	☒
	iva_es_costo	char(1)	☒
	cantidad_promocion_...	real	☒
	fecha_inicia_promocion	datetime	☒
	grupo_comision	varchar(10)	☒
	calificacion_abc	char(1)	☒
	fec_cambioPrecio_vta	smalldatetime	☒
	tipo_1	varchar(20)	☒
	tipo_2	varchar(20)	☒
	tipo_3	varchar(20)	☒
	tipo_4	varchar(20)	☒
	tipo_5	varchar(20)	☒
	tipo_6	varchar(20)	☒
	tipo_7	varchar(20)	☒
	precio_pos	money	☒
	id	int	☐
	calificacion_abc_ant	varchar(1)	☒
	no_aplica_retenciones	char(1)	☒
	accesorio	char(1)	☒
	fecha_actualizacion	datetime	☒
	si_flota	bit	☒
	abc_tendencia	varchar(2)	☒
	num_fabricacion	varchar(30)	☒
	Notas	varchar(500)	☒
	es_flota	char(1)	☒
	horas_uso	real	☒
	tipo_8	int	☒
	tipo_9	int	☒
	MinDespacho	int	☒
	restringir_pos	char(1)	☒
	esLubricante	bit	☒
	EsPintura	bit	☒
	esNeumatico	bit	☒
			☐

Figura 3-6 Tabla Referencias (Montalvo, 2019)

En la Figura 3-6 se puede ver a mayor detalle la tabla referencias, esta tabla contiene todo el inventario de partes, accesorios y vehículos, tanto nuevos como vehículos que ingresan al taller y que no fueron vendidos por la empresa. La clave principal de esta tabla es el código que es un varchar de 20, esta tabla también se relaciona con otras para permitir la clasificación y control del inventario esta estructura se revisará más adelante a la fecha de migración esta tabla tenía 139097 registros.

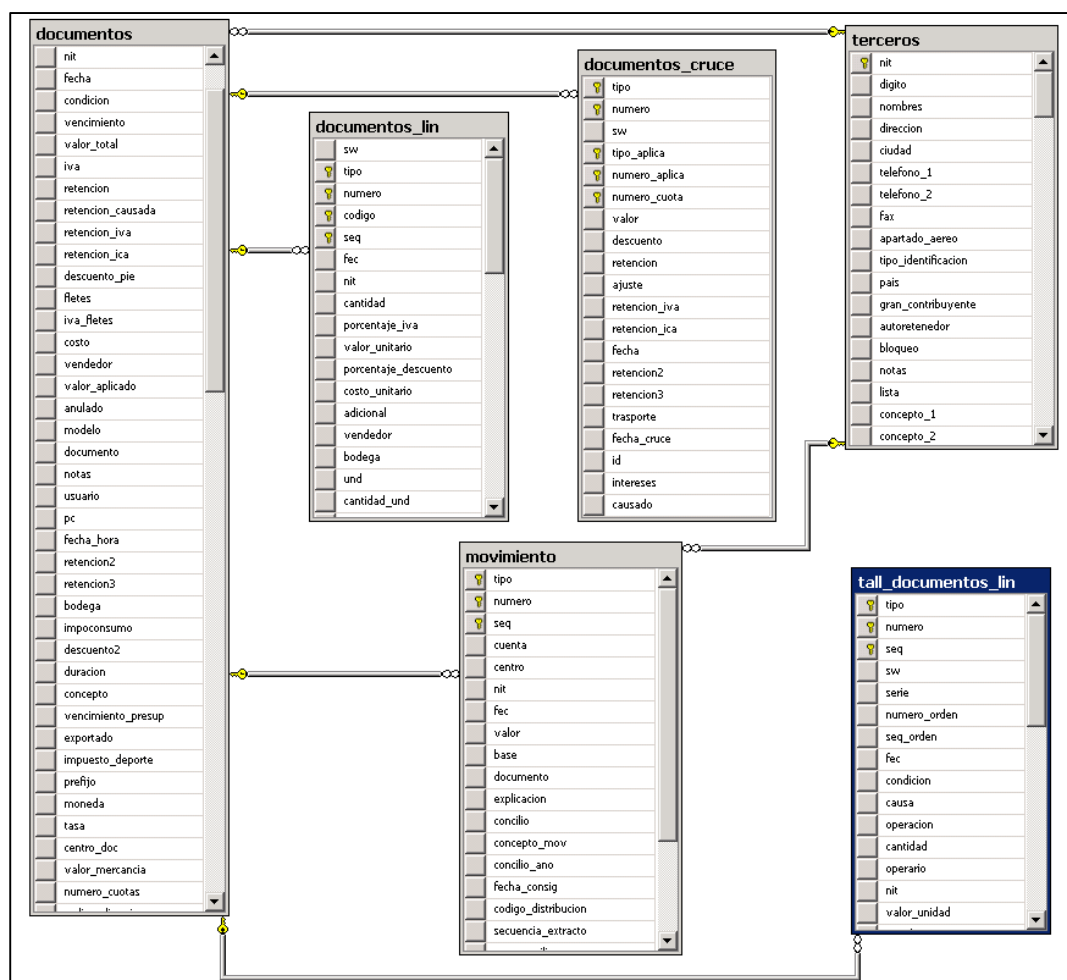


Figura 3-7 Modelo Entidad Relación, Contabilidad y Modulo (Montalvo, 2019)

En la figura 3-7 se presenta el modelo con el que el ERP DMS, gestiona la información contable y la información de transacciones como factura, notas de crédito, retenciones, etc. Este modelo tiene también dos tablas muy importantes que se las analizarán a detalle más adelante. Como son la tabla documentos y la tabla movimiento, se puede ver además que aquí también está relacionada la tabla de terceros que ya se trató anteriormente.

Por otro lado, este diagrama tiene una relación con el diagrama de talleres, que se revisó anteriormente, mediante la tabla `tall_documentos_lin`. En resumen, la tabla `documentos` almacenará el encabezado de los documentos que se generen, la tabla `documentos_lin` las líneas o el detalle de estos documentos, la tabla `movimiento` en cambio genera la información de los asientos contables de estos documentos, la tabla `documentos_cruce` las aplicaciones o pagos con los que se cancelan los documentos y por último la tabla `tall_documento_lin` la información o relación con las ordenes de taller y sus respectivas facturas, o notas de crédito.

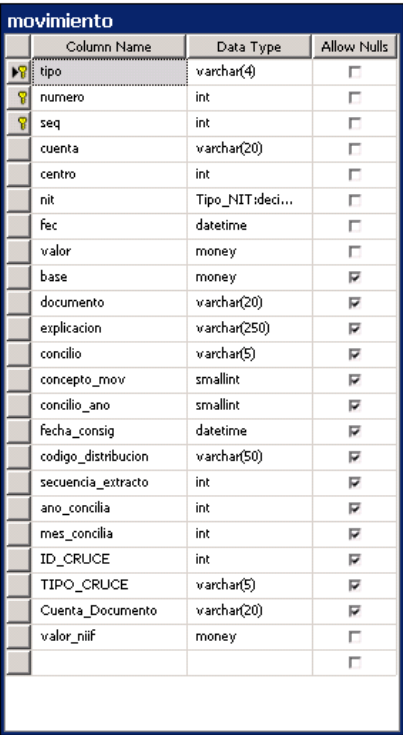
documentos			
	Column Name	Data Type	Allow Nu ▲
	sw	tinyint	☐
Y	tipo	varchar(4)	☐
Y	numero	int	☐
	nit	Tipo_NIT:deci...	☐
	fecha	datetime	☐
	condicion	varchar(4)	☒
	vencimiento	datetime	☒
	valor_total	money	☒
	iva	money	☒
	retencion	money	☒
	retencion_causada	money	☒
	retencion_iva	money	☒
	retencion_ica	money	☒
	descuento_pie	money	☒
	fletes	money	☒
	iva_fletes	money	☒
	costo	money	☒
	vendedor	Tipo_NIT:deci...	☒
	valor_aplicado	money	☒
	anulado	bit	☐
	modelo	varchar(5)	☒
	documento	varchar(20)	☒
	notas	varchar(500)	☒
	usuario	varchar(10)	☐
	pc	varchar(20)	☐
	fecha_hora	datetime	☐
	retencion2	money	☒
	retencion3	money	☒
	bodega	smallint	☐
	impoconsumo	money	☒
	descuento2	money	☒
	duracion	smallint	☒
	...	...	...

Figura 3-8 Tabla Documentos (Montalvo, 2019)

En la Figura 3-8 se representa la tabla `documentos` cuyos primary Key son los campos `tipo` un `varchar` de longitud de 4 y `numero` un `int`, además de tener campos con el `nit` que nos permite relacionar un documento a una persona, tiene también el campo `fecha` que es un `datetime`, así como también el campo `valor_total`, el campo `iva`, el campo `valor_aplicado`, y el campo `costo`, todos ellos de tipo `money`, que almacenaran los valores

de cada documento de manera general. Esta tabla cuenta con 3018247 registros siendo una de las más grandes dentro de la base de datos. Una de las particularidades de esta tabla es que también maneja el campo sw, que también estaba presente en una de las tablas del modelo de vehículos, este campo permite de manera rápida poder segmentar la información por tipo de documento, es decir, si el documento es una venta, una compra, una devolución en ventas, o una devolución en compras, etc.

Por otro lado en esta tabla también se almacena información del usuario y el nombre del computador desde donde se generó la transacción siendo esta información muy útil para temas de auditoría.



	Column Name	Data Type	Allow Nulls
PK	tipo	varchar(4)	☐
PK	numero	int	☐
PK	seq	int	☐
	cuenta	varchar(20)	☐
	centro	int	☐
	nit	Tipo_NIT:deci...	☐
	fec	datetime	☐
	valor	money	☐
	base	money	☒
	documento	varchar(20)	☒
	explicacion	varchar(250)	☒
	concilio	varchar(5)	☒
	concepto_mov	smallint	☒
	concilio_ano	smallint	☒
	fecha_consig	datetime	☒
	codigo_distribucion	varchar(50)	☒
	secuencia_extracto	int	☒
	ano_concilia	int	☒
	mes_concilia	int	☒
	ID_CRUCE	int	☒
	TIPO_CRUCE	varchar(5)	☒
	Cuenta_Documento	varchar(20)	☒
	valor_niif	money	☐
			☐

Figura 3-9 Tabla Movimiento (Montalvo, 2019)

En la figura 3-9 se observa la tabla movimiento la misma que contiene como clave principal o primary key los campos tipo, numero y seq, estos campos almacenan la información que relaciona los asientos o registros contables con cada documento que se genera en el ERP. El campo seq permite que se tenga repetido los campos tipo y numero, es decir un documento tiene afectación en varias cuentas contables pero como se registra la información en esta tabla para no perder la integridad es con el campo seq que vendría a ser la secuencia de cada registro.

En esta tabla también se almacena el nit campo que sirve para relacionar la misma con la tabla terceros que es la que contiene la información de las personas. Además, esta tabla tiene el campo fec, que es de tipo datetime y el campo valor, este campo para diferencias cuentas de debe y de haber se almacena con el signo positivo o negativo, dependiendo de la naturaleza de la cuenta, esta tabla cuenta con 11129001 registros siendo así como la tabla documentos una de las más grandes.

5. CAPÍTULO IV: ERP KAIROS

En el presente capítulo se pretende analizar de manera general el ERP¹¹ Kairós, sus generalidades, lenguaje de programación que utiliza, con qué DBMS trabaja, la empresa que lo desarrollo, su implementación en PROAUTO C.A., y actualmente con que infraestructura está trabajando. También se revisará a profundidad el modelo entidad relación de las principales tablas que se afectan en la transaccionalidad diaria de la compañía.

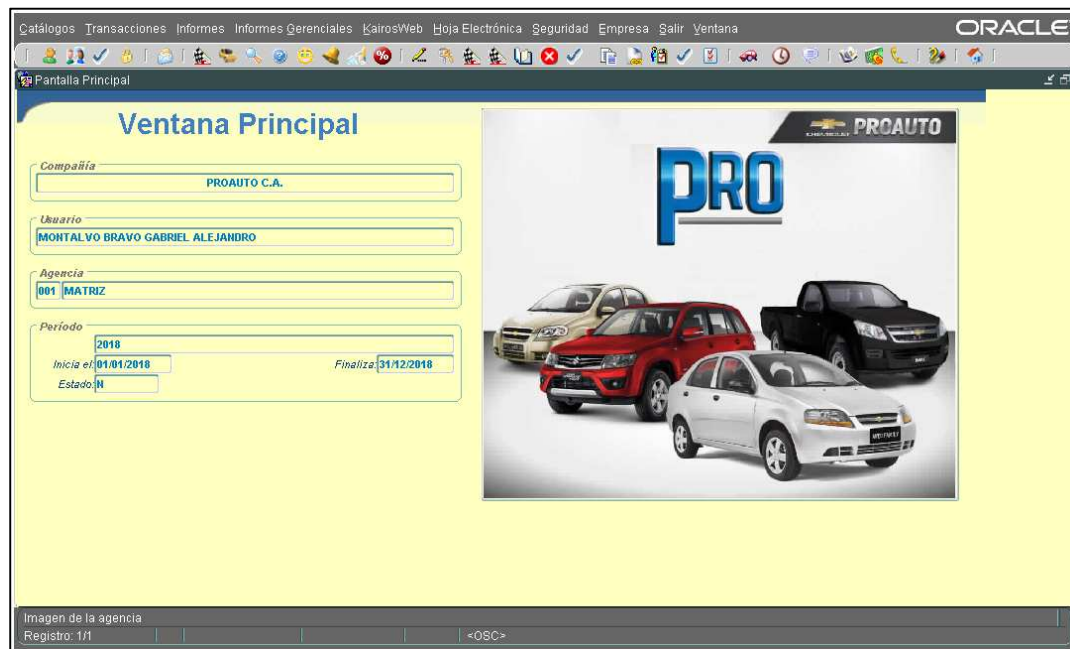


Figura 4-1 Pantalla Principal Kairós (Montalvo, 2019)

La Figura 4-1 representa la pantalla principal de los usuarios la misma que está dividida en secciones como: menú principal, botones de accesos directos y ventana principal en donde se desplegarán los diferentes módulos que dispone el sistema. Además de tener la información de la agencia en la que se trabaja también se encuentra el nombre del usuario que ha iniciado sesión.

5.1. Generalidades

¹¹ ERP: sistemas de planificación de recursos empresariales en inglés, enterprise resource planning

Kairos ERP es un sistema que está desarrollado con tecnología Oracle, sus fuentes son generados a partir de una herramienta case que es Oracle Designer, y funcionan con Oracle Forms and Reports en la versión 10g, mientras que se soporta en el gestor de base de datos Oracle Standard Edition 11g.

Oracle Designer, anteriormente llamado Designer/2000 y Oracle Case, es una herramienta CASE de modelado, utilizada para analizar requisitos de negocios, así como para diseñar y generar sistemas cliente/servidor que puedan satisfacer tales requisitos. También se le denomina herramienta de documentación, en tanto la función de las herramientas de diseño de bases de datos es almacenar toda la información relativa al diseño en un único lugar. En sus orígenes, cuando fue lanzado al mercado por Oracle, en 1988, recibía el nombre de CASE Designer, en 1998 Case Designer fue renombrado a Designer/2000, desapareciendo así mismo el sufijo 2000 con el paso del tiempo. (SCRIBD, 2018)

SoftSierra S.A. es la empresa que desarrolló este ERP, esta empresa tiene su sede en la ciudad de Ambato, cuenta con más de 15 años en el mercado. Este ERP actualmente tiene muy buena acogida entre concesionarios de vehículos de distintas marcas como son Hyundai, Volkswagen, y Chevrolet, además de tener presencia también en el Perú, en concesionarios de vehículos de la marca Chevrolet y Honda.

En mayo del 2017 PROAUTO C.A. empieza a trabajar con este sistema como su principal ERP. Luego de un proceso de migración de data que duró aproximadamente 6 meses y después de un año de uso, el sistema ha sido estabilizado. Kairos, es un sistema modular cuya interfaz de acceso al usuario final es a través de un navegador Web, como funciona con Forms and Reports de Oracle, su limitante es que este navegador debe ser Internet Explorer.

Como requerimiento para su funcionamiento es la instalación del KVM¹² y funciona con dependencia de Java. Este ERP cuenta con módulos para el manejo de venta de vehículos, administración de talleres, venta y reserva de repuestos por mostrador, manejo de inventarios módulos de cartera y crédito, y además de todo el manejo contable de la compañía.

¹² KVM: La máquina virtual K (en inglés K virtual machine, KVM) es una máquina virtual desarrollada por Sun Microsystems

Los reportes son generados a través del mismo sistema con la utilización de Reports de Oracle, estos reportes sin embargo no tienen la opción a ser personalizados por el administrador del sistema, sino que son de mantenimiento y mejoras por parte de la compañía desarrolladora del Software. Sin embargo, para suplir esta necesidad el sistema cuenta con un acceso para creación de consultas SQL que se integran al menú del sistema y su resultado es extraído directamente en una hoja de cálculo.

El Sistema gestor de base de datos que soporta este ERP es Oracle Db 10g, con un total de 1579 tablas en las que están incluidas las tablas de auditoria tipo Journal, es decir el total de tablas que almacenan los datos sensibles serian la mitad 790, 315 vistas, 204 Procedimientos Almacenados y 154 funciones que almacenan y gestionan la transaccionalidad y reportería del sistema Kairos.

En la actualidad la base de datos tiene una capacidad ocupada en disco de 35.5 Gigas correspondiente a data transaccional desde marzo 13 del 2017, con corte 5 de noviembre del 2018 y data migrada desde junio del 2006 a marzo 12 del 2018. Es importante notar esta diferencia ya que las tablas de journal de Oracle tienen únicamente auditoría de la data transaccional es decir data que ha sido generada desde el ERP.

5.2. Infraestructura

En la salida a producción del ERP Kairos, disponía de un equipo HP ProLiant Gen8, con 32 GB de memoria RAM y un Procesador Intel, y dos discos duros de 600GB SAS de 10K. En este equipo estaba instalado VMWare Esxi 6.0 como virtualizador y sobre este Virtualizador se tenía un Servidor Virtual que tenía como sistema Operativo Centos 4.6 y la instalación de Forms and Reports 10g de Oracle. Este equipo era el servidor de aplicaciones. Por otro lado, se decidió utilizar el mismo servidor que alojaba el sistema anterior DMS, pero también en el mismo esquema de Virtualización con VMWare Esxi 6.0 como virtualizador.

Actualmente el ERP Kairós trabaja sobre una plataforma virtualizada con OVM, en un servidor físico HP ProLiant ML350 Gen9 con 192 GB de memoria RAM distribuida en 96GB para cada procesador, dos procesadores Intel(R) Xeon(R) CPU E5-2650 v3 @ 2.30GHz con discos 6 discos duros, 2 de 450GB SAS 15k, 2 de 600GB SAS 10K y 2 más de 600GB SAS 10k, en arreglos de la siguiente manera: los discos de 450 en RAID

1. Este servidor tiene una máquina Virtual con 64GB de memoria RAM, asignados 16 Cores Virtuales, y espacio en disco duro de 1Tb, su sistema operativo es Oracle Linux 6 de 64 bits, aquí está instalado Oracle Db 11g versión estándar siendo este el servidor de base de datos. Sobre este mismo servidor físico también está virtualizado un segundo servidor con 32 Gb de memoria RAM, asignados 8 Cores Virtuales y espacio en disco de 400gb, su sistema operativo es Red Hat Linux 4.8 de 64 bits, aquí se tiene instalado Forms and Reports 10g, siendo este equipo el servidor de aplicaciones.

5.3. DBMS con el que Trabaja

Kairos ERP, nació desarrollado en herramientas Oracle, por esta razón su gestor de base de datos es Oracle. Al inicio de su desarrollo empezaron con la versión de Oracle db 8g, sin embargo, durante el tiempo han ido migrando hasta llegar a la versión que actualmente está instalada en PROAUTO C.A. que es la versión Oracle db 11g.

5.3.1. Modelo Entidad Relación en ORACLE

Kairos cuenta con un modelo entidad relación que maneja alrededor de 1500 tablas, este capítulo pretende analizar las tablas más importantes y además el modelo entidad relación de algunas de ellas enfocados en los procesos con los que se opera el área comercial y el área de talleres en PROAUTO C.A.

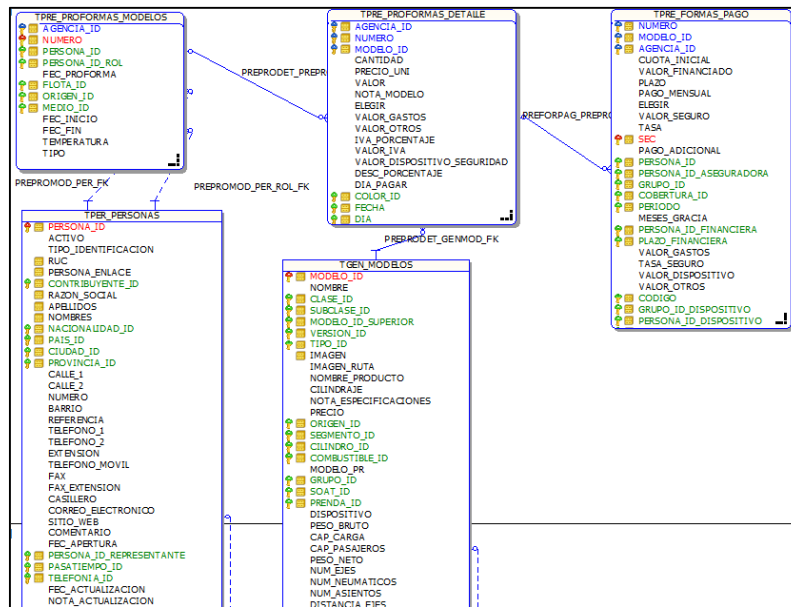


Figura 4-2 Modelo Entidad Relación Cotizador Vehicular (Montalvo, 2019)

En la Figura 4-2 se observa el modelo entidad relación con el que el ERP administra la generación de una cotización de vehículos. En éste intervienen tablas importantes como tper_personas, la misma que almacena toda la información de contacto de personas, la tabla tpre_proformas_modelos que es el encabezado de la proforma o cotización, esta tabla tiene una relación de uno a muchos con la tabla tpre_proformas_detalle donde se almacenan los datos del modelo del vehículo y la información de los valores como precio unitario, iva, descuentos, etc.. Por otro lado está la tabla tgen_modelos, en esta tabla está la información de los modelos de vehículos que van a ser cotizados y por último se puede ver la tabla tpre_formas_pago, la misma que almacena las distintas opciones de pago que podría tener una cotización si es a crédito o de contado.

TPER_PERSONAS	
PERSONA_ID	VARCHAR2(20 Byte) NN (PK) (D10)
ACTIVO	VARCHAR2(1 Byte)
TIPO_IDENTIFICACION	VARCHAR2(13 Byte) NN (D15)
ROL	VARCHAR2(20 Byte) NN (AK1,D18)
PERSONA_ENLACE	VARCHAR2(20 Byte) NN (FK) (D12)
CONTRIBUYENTE_ID	VARCHAR2(2000 Byte) NN (D15)
RAZON_SOCIAL	VARCHAR2(50 Byte) NN (D15)
APELLIDOS	VARCHAR2(50 Byte) NN (D15)
NOMBRES	VARCHAR2(50 Byte) NN (D15)
NACIONALIDAD_ID	VARCHAR2(3 Byte) NN (FK) (D17)
PAIS_ID	VARCHAR2(4 Byte) NN (FK) (D13)
CIUDAD_ID	VARCHAR2(4 Byte) NN (FK) (D13)
PROVINCIA_ID	VARCHAR2(4 Byte) NN (FK) (D13)
CAJUE_1	VARCHAR2(60 Byte) NN
CAJUE_2	VARCHAR2(60 Byte) NN
NUMERO	VARCHAR2(10 Byte) NN
BARRIO	VARCHAR2(60 Byte) NN
REFERENCIA	VARCHAR2(2000 Byte) NN
TELEFONO_1	VARCHAR2(15 Byte) NN
TELEFONO_2	VARCHAR2(15 Byte) NN
EXTENSION	VARCHAR2(10 Byte) NN
TELEFONO_MOVIL	VARCHAR2(15 Byte) NN
FAX	VARCHAR2(15 Byte) NN
FAX_EXTENSION	VARCHAR2(10 Byte) NN
CASILLERO	VARCHAR2(10 Byte) NN
CORREO_ELECTRONICO	VARCHAR2(200 Byte) NN
SITIO_WEB	VARCHAR2(200 Byte) NN
COMENTARIO	VARCHAR2(400 Byte) NN
FEC_APERTURA	DATE NN
PERSONA_ID_REPRESENTANTE	VARCHAR2(20 Byte) (FK) (D16)
PERSONA_ID_REPRESENTANTE	VARCHAR2(20 Byte) (FK) (D16)
TELEFONO_A_ID	VARCHAR2(10 Byte) (FK) (D11)
FEC_ACTUALIZACION	DATE
NOTA_ACTUALIZACION	VARCHAR2(200 Byte) NN
ENCUESTABLE	VARCHAR2(1 Byte) NN
COOPERATIVA_ID	VARCHAR2(20 Byte) (FK) (D11)
ACTUALIZABLE	VARCHAR2(1 Byte) NN
ACTIVIDAD	VARCHAR2(200 Byte) NN
CARTAL_SOCIAL	NUMBER(18,4)
FEC_CONSTITUCION	DATE
CUOTA_INGRESO	NUMBER(12,2)
CUOTA_MENSUAL	NUMBER(12,2)
FEC_INACTIVO	DATE
FEC_REACTIVACION	DATE
MATRICULA_COMERCIO	VARCHAR2(20 Byte) NN
FEC_MATRICULACION	DATE
FEC_COSTITUCION	DATE
ALIAS	VARCHAR2(100 Byte) (FK) (D16)
CARGO_ID	VARCHAR2(3 Byte) NN
ENVIOS_BIT	VARCHAR2(1 Byte) NN
ESTADO_PER	VARCHAR2(1 Byte) NN
DINARDAP	VARCHAR2(1 Byte) NN
DIR_SERVICO_BASICO	VARCHAR2(200 Byte) NN

Figura 4-3 Tabla tper_personas (Montalvo, 2019)

En la Figura 4-3 se puede observar con mejor detalle la tabla tper_personas, la nomenclatura que maneja Kairos para la creación de las tablas es la letra “t” determina que se trata de una tabla, las siguientes 3 letras “per” hacen referencia a que se relaciona con tablas de personas y por último es el nombre de la información que se almacena en dicha tabla. La clave principal de esta tabla es el campo persona_id, que es un varchar de 20. Además como se puede observar que esta tabla maneja una referencia recursiva, tiene todos los campos de contactabilidad de personas, se almacenan clientes, empleados, proveedores, etc., es decir toda persona que tiene relación con PROAUTO C.A..

Para distinguir las diferencias entre cada uno de estos, el sistema maneja una tabla adicional que no está diagramada, que se llama tper_rols, en donde se categorizan de acuerdo al rol que tiene la persona dentro del sistema. Por ejemplo cliente, proveedor, vinculándose con el campo persona_id, es decir la relación entre estas dos tablas es de uno a muchos. Una persona podría tener más de un rol, esta tabla tiene 131612 registros con corte 11 de diciembre del 2018.

TGEN_MODELOS			
MODELO_ID	VARCHAR2 (20 Byte)	NN (PK)	(D1,2)
NOMBRE	VARCHAR2 (100 Byte)	NN	
CLASE_ID	VARCHAR2 (10 Byte)	(FK)	(D16)
SUBCLASE_ID	VARCHAR2 (10 Byte)	(FK)	(D1,1)
MODELO_ID_SUPERIOR	VARCHAR2 (20 Byte)	(FK)	(D16)
VERSION_ID	VARCHAR2 (10 Byte)	(FK)	(D2)
TIPO_ID	VARCHAR2 (10 Byte)	(FK)	(D9)
IMA_GEN	BLOB		(D4)
IMA_GEN_RUTA	VARCHAR2 (200 Byte)		
NOMBRE_PRODUCTO	VARCHAR2 (60 Byte)		
CILINDRAJE	VARCHAR2 (10 Byte)		
NOTA_ESPECIFICACIONES	VARCHAR2 (4000 Byte)		
PRECIO	NUMBER (18,4)		
ORIGEN_ID	VARCHAR2 (6 Byte)	NN (FK)	(D10)
SEGMENTO_ID	VARCHAR2 (3 Byte)	NN (FK)	(D3)
CILINDRO_ID	VARCHAR2 (3 Byte)	NN (FK)	(D7)
COMBUSTIBLE_ID	VARCHAR2 (3 Byte)	NN (FK)	(D4)
MODELO_PR	VARCHAR2 (20 Byte)		
GRUPO_ID	VARCHAR2 (3 Byte)	(FK)	(D5)
SOAT_ID	VARCHAR2 (3 Byte)	(FK)	(D1)
PRENDA_ID	VARCHAR2 (3 Byte)	(FK)	(D1,3)
DISPOSITIVO	VARCHAR2 (1 Byte)	NN	
PESO_BRUTO	NUMBER (10,2)		
CAP_CARGA	VARCHAR2 (40 Byte)		
CAP_PASAJEROS	VARCHAR2 (40 Byte)		
PESO_NETO	NUMBER (18,6)		
NUM_EJES	NUMBER (12)		
NUM_NUMMATICOS	NUMBER (12)		
NUM_ASIENOS	NUMBER (12)		
DISTANCIA_BES	NUMBER (18,6)		
LARGO	NUMBER (18,6)		
ALTURA	NUMBER (18,6)		
POTENCIA	VARCHAR2 (60 Byte)		
TRACCION	VARCHAR2 (1 Byte)		
PESO_CARROZADO	NUMBER (18,6)		
NUM_PUERTAS	NUMBER (12)		
LONGITUD_PROFUNDIDAD	NUMBER (12,4)		
LONGITUD_ALTO	NUMBER (12,4)		
LONGITUD_ANCHO	NUMBER (12,4)		
NUM_POLIZA	VARCHAR2 (50 Byte)		
TRANSMISION	VARCHAR2 (1 Byte)		
ARCHIVO_ESPECIFICACION	VARCHAR2 (100 Byte)		
AÑO	NUMBER (4)		
AÑO_FABRICACION	NUMBER (4)		
DIAS_INV_VERDE	NUMBER (4)		
DIAS_INV_ROJO	NUMBER (4)		
IMA_GEN_360_EXT	VARCHAR2 (200 Byte)		
IMA_GEN_360_INT	VARCHAR2 (200 Byte)		
VIDEO	VARCHAR2 (200 Byte)		
ACCESORIOS	VARCHAR2 (1 Byte)		

Figura 4-4 Tabla tgen_modelos (Montalvo, 2019)

En la Figura 4-4 se observan los campos que conforman la tabla tgen_modelos las letras “gen” que conforman el nombre significan general es decir esta tabla está dentro de las tablas de información general que maneja Kairos. La clave principal o primary key de esta tabla es el campo modelo_id el mismo que es un varchar de 20 caracteres así como la tabla de personas. Esta tabla también maneja una relación recursiva con el campo modelo_id_superior, esta relación permite agrupar a los distintos modelos en familias de modelos por ejemplo: si se habla de un Aveo, en este modelo existen varias versiones sin embargo para los análisis de ventas se podría generar información solo de la familia AVEO. Esta tabla almacena toda la información de los modelos de vehículos, esta información sirve por ejemplo para la matriculación. Tiene campos como el peso_bruto con tipo de dato number, cilindraje varchar de 10, cap_pasajeros varchar(40), además de información para clasificar el tipo de vehículo si es pasajeros, carga, suv, etc., vinculándose con otras tablas mediante los campos clase_id, subclase_id. Esta tabla tiene 1023 registros con corte al 11 de diciembre del 2018.



50

como costo, precio, tiempo_estandar, descuento_tot, y para repuestos campos como cantidad, precio, descuento, entregado, fecha. Por último, la información del vehículo asociado a la orden de trabajo se almacena en la tabla ttal_registro_vehiculos, aquí se tiene una particularidad y es que esta tabla es propia para el manejo de talleres, no tiene relación con la información de vehículos nuevos, su clave principal es el número de chasis.

TTAL_ORDENES_TRABAJO			
AGENCIA_ID	VARCHAR2 (3 Byte)	NN (PK)	(IX11,IX16)
NUM_ORDEN_TALLER	INTEGER	NN (PK)	(IX16)
PERSONA_ID	VARCHAR2 (20 Byte)	NN (FK)	(IX15)
CHASIS	VARCHAR2 (20 Byte)	NN (FK)	(IX4)
TIPO_ORDEN_ID	VARCHAR2 (2 Byte)	NN (FK)	(IX6)
KILOMETRAJE	INTEGER	NN	
COMBUSTIBLE	VARCHAR2 (1 Byte)	NN	
PARQUEO	VARCHAR2 (5 Byte)	NN	
FEC_RECEPCION	DATE	NN	
ESTADO	VARCHAR2 (1 Byte)	NN	
PERSONA_ID_ASESOR	VARCHAR2 (20 Byte)	(FK)	(IX14)
AGENCIA_ID_RESERVA	VARCHAR2 (3 Byte)	(FK)	(IX1)
RESERVA_ID	INTEGER	(FK)	(IX3)
PERSONA_ID_CALIDAD	VARCHAR2 (20 Byte)	(FK)	(IX9)
FEC_PROMETIDA	DATE		
FEC_ENTREGA	DATE		
OBSERVACION	VARCHAR2 (1500 Byte)		
NOTA_CLIENTE	VARCHAR2 (1500 Byte)		
NOTA_ASESOR	VARCHAR2 (1500 Byte)		
FIRMA_CALIDAD	VARCHAR2 (1 Byte)		
FIRMA_FINALIZADA	VARCHAR2 (1 Byte)		
PERSONA_ID_FINALIZAR	VARCHAR2 (20 Byte)	(FK)	(IX5)
SEC_ORDEN	NUMBER (30)	NN	
CONO	VARCHAR2 (10 Byte)		
LLANTA_ID	VARCHAR2 (20 Byte)	(FK)	(IX2)
ESTADO_ID	VARCHAR2 (10 Byte)	(FK)	(IX10)
FEC_FIN_TAREAS	DATE		
MEDIDA_ID	VARCHAR2 (20 Byte)	(FK)	(IX7)
FEC_CONTROL	DATE		
FEC_FINALIZACION	DATE		
FEC_FACTURACION	DATE		
AGENCIA_ID_RES	VARCHAR2 (3 Byte)	(FK)	(IX3)
NUM_ORDEN_TALLER_RES	INTEGER	(FK)	(IX3)
PERSONA_ID_RECIBE	VARCHAR2 (20 Byte)	(FK)	(IX12)
SEC_ORDEN_ENTREGA	NUMBER (30)		
TAXI	VARCHAR2 (1 Byte)	NN	
AUDITAR	VARCHAR2 (1 Byte)	NN	
NOTIFICADO	VARCHAR2 (1 Byte)	NN	
FACTURADO	VARCHAR2 (1 Byte)		
SINIESTRO	VARCHAR2 (50 Byte)		
FEC_ACTUALIZACION	DATE		
PERSONALIZADO	VARCHAR2 (1 Byte)		
PERSONA_ID_ENTREGA	VARCHAR2 (20 Byte)	(FK)	(IX8)
FEC_INGRESO	DATE		
MOTIVO_ID	VARCHAR2 (3 Byte)	(FK)	(IX13)
TIPO_SERVICIO	VARCHAR2 (20 Byte)		

Figura 4-6 Tabla Ttal_ordenes_trabajo (Montalvo, 2019)

La Figura 4-6 representa el modelo de la tabla ttal_ordenes_trabajo, se puede observar que su clave principal o primary key está conformada por los campos agencia_id que es un varchar de 3, el mismo que almacena el número de identificación de la agencia donde se generó la orden y el campo num_orden_taller un campo de tipo integer, que almacena el número de la orden de trabajo. Por otro lado esta tabla maneja también datos importantes como por ejemplo el campo kilometraje de tipo integer que almacena el kilometraje con el que el vehículo ingreso al taller. Este dato sirve para cálculos estadísticos y poder determinar cuándo sería el próximo ingreso del vehículo al taller. Otros campos importantes en esta tabla son por ejemplo los campos persona_id, chasis y tipo_orden_id que son claves foráneas con tablas importantes como tper_personas y ttal_registro_vehiculos, respectivamente. Se puede ver además que se manejan varios campos con las letras “fec” que hacen referencia a fechas de momentos importantes

durante el proceso de una orden de taller todos estos campos son de tipo date, con fecha 12 de diciembre de 2018 esta tabla tiene 405214 registros.

TINV_PRODUCTOS			
PRODUCTO_ID	VARCHAR2(20 Byte)	NN (FK)	(1018)
BODEGA_ID	VARCHAR2(3 Byte)	NN (FK)	(108)
CATEGORIA_ID	VARCHAR2(10 Byte)	NN (FK)	(105)
NOMBRE	VARCHAR2(60 Byte)	NN	
CALIFICACION	VARCHAR2(1 Byte)	NN	
IVA_PORCENTAJE	NUMBER(52)	NN	
COLOR_ID	VARCHAR2(10 Byte)	(FK)	(1017)
PESO_ID	VARCHAR2(3 Byte)	(FK)	(1019)
VERSION_ID	VARCHAR2(10 Byte)	(FK)	(109)
PERIODO	NUMBER(4)		
CILINDRAJE	VARCHAR2(10 Byte)		
PRECIO_BLOQUEADO	VARCHAR2(1 Byte)	NN	
PEC_APERTURA	DATE	NN	
LONGITUD_ID	VARCHAR2(3 Byte)	(FK)	(101)
MARCA_ID	VARCHAR2(10 Byte)	(FK)	(1021)
VOLUMEN_ID	VARCHAR2(3 Byte)	(FK)	(1022)
SERIE	VARCHAR2(20 Byte)		
MOTOR	VARCHAR2(20 Byte)		
CHASIS	VARCHAR2(20 Byte)		
COSTO_FOB	NUMBER(18,4)	NN	
COSTO_CIF	NUMBER(18,4)	NN	
CADUCA	VARCHAR2(1 Byte)	NN	
CLASE_ID	VARCHAR2(10 Byte)	(FK)	(1016)
MODELO_ID	VARCHAR2(20 Byte)	(FK)	(1021)
PARTIDA_ID	VARCHAR2(15 Byte)	(FK)	(1025)
SUBCLASE_ID	VARCHAR2(10 Byte)	(FK)	(1015)
TIPO_ID	VARCHAR2(10 Byte)	(FK)	(106)
UNIDAD_ID	VARCHAR2(3 Byte)	(FK)	(1012)
PESO_BRUTO	NUMBER(12,6)		
PESO_NETO	NUMBER(12,6)		
LONGITUD_ALTO	NUMBER(12,4)		
LONGITUD_ANCHO	NUMBER(12,4)		
LONGITUD_PROFUNDIDAD	NUMBER(12,4)		
VOLUMEN	NUMBER(12,4)		
APILAR	NUMBER(8)		
CARACTERISTICA	VARCHAR2(1000 Byte)		
NOTA_PROMOCION	VARCHAR2(2000 Byte)		
NOTA_ESPECIFICACIONES	VARCHAR2(4000 Byte)		
NOTA_COMPRAS	VARCHAR2(4000 Byte)		
NOTA_VENTAS	VARCHAR2(4000 Byte)		
CODIGO_ID	VARCHAR2(3 Byte)	NN (FK)	(1023)
COMPARTIMIENTO_ID	VARCHAR2(3 Byte)	NN (FK)	(104)
STATUS_ID	VARCHAR2(1 Byte)	NN (FK)	(1011)
ANALIZAR_MOV	VARCHAR2(1 Byte)	NN (FK)	(1011)
CAP_CARGA	VARCHAR2(40 Byte)		
CAP_PASAJEROS	VARCHAR2(40 Byte)		
PRECIO_SUGERIDO	NUMBER(18,4)		
PRODUCTO_BC	VARCHAR2(20 Byte)	(AND AK1, AK3, 105, 1020)	
PRODUCTO_PR	VARCHAR2(20 Byte)	(AND AK1, AK4, 107, 1020)	
PROMOCION	VARCHAR2(1 Byte)		
RIN	NUMBER(52)		
MEDIDA_ID	VARCHAR2(20 Byte)	(FK)	(1010)
DISEÑO_ID	VARCHAR2(10 Byte)	(FK)	(1010)
LLANTA_ID	VARCHAR2(20 Byte)	(FK)	(1014)
ESTADO_ID	VARCHAR2(10 Byte)	(FK)	(1013)
LLANTA_TIPO	VARCHAR2(1 Byte)		
PLACA	VARCHAR2(10 Byte)		
ENVIO_EXT	VARCHAR2(1 Byte)		
TEST	VARCHAR2(1 Byte)		
DESDE	DATE		
HASTA	DATE		
DESCUENTO_MAXIMO	NUMBER(52)		
PRECIO_ID	VARCHAR2(3 Byte)	(FK)	(102)
COLOR_ID_INTERIOR	VARCHAR2(10 Byte)	(FK)	(1014)
PESO_CARROZADO	NUMBER(18,6)		
NUM_ES	NUMBER(12)		
NUM_PASAJEROS	NUMBER(12)		

Figura 4-7 Tabla tinv_productos (Montalvo, 2019)

En la Figura 4-7 se observa la tabla tinv_productos cuya clave principal o primary key es el campo producto_id, con tipo de dato varchar de 20, las letras del nombre “inv” hacen referencia a inventario, en esta tabla se almacenan la información del manejo de inventario. Tanto repuestos como accesorios y vehículos nuevos, por esta razón en esta tabla se encuentran campos como serie varchar de 20, chasis varchar de 20, motor varchar de 20, entre otros campos que hacen referencia a la información de un vehículo.

Kairos para la clasificación de ítems dentro de esta tabla utiliza un concepto de bodegas con una tabla que no se ve en este análisis que es la tabla tinv_bodegas en esta tabla los campos más relevantes son bodega_id, línea_id, nombre, negativo, esta tabla se relaciona con tinv_productos mediante el campo bodega_id, el objetivo de esta tabla es agrupar por un código los diferentes tipos de productos como por ejemplo: baterías que tiene una bodega, lubricantes manejan otra bodega, vehículos livianos, vehículos pesados, entre

otros. Es por lo que esta es una de las tablas más importantes dentro del sistema con 100552 registros al 18 de diciembre del 2018.

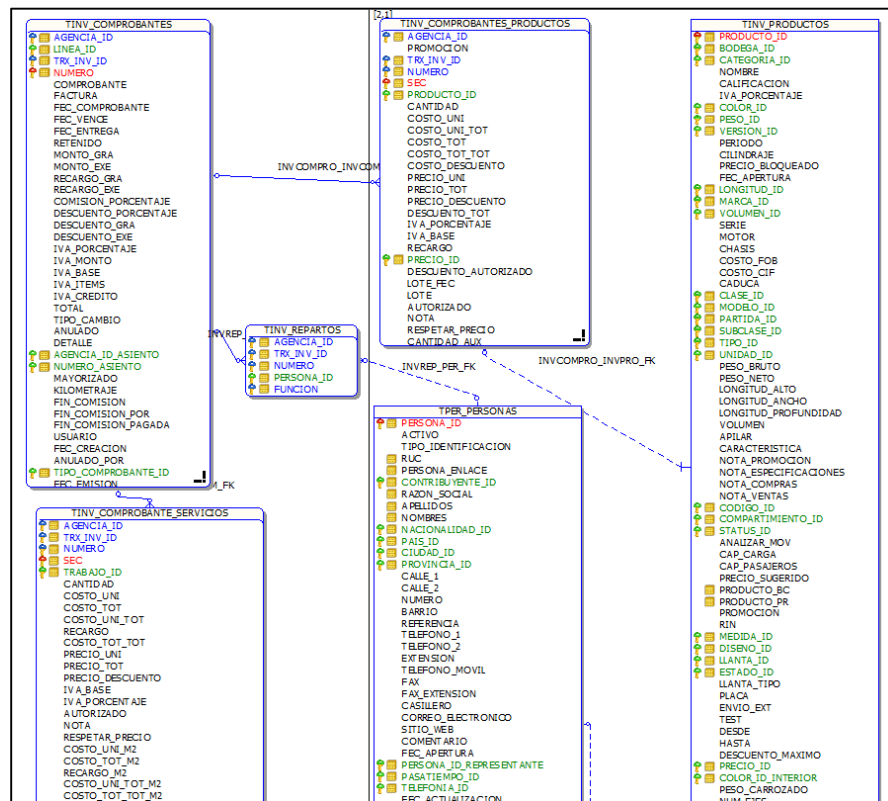


Figura 4-8 Modelo Entidad Relación Comprobantes (Montalvo, 2019)

En la Figura 4-8 se puede observar el modelo entidad relación de Kairos, como administra la información que se genera en el manejo de comprobantes, es decir facturas, notas de crédito, altas y bajas de inventario, etc., es importante mencionar que en estas tablas no se registran retenciones ya que este tipo de documento es considerado como un asiento, estructura que se revisará más adelante. Las tablas que componen este modelo son *tinvs_comprobantes* que es la tabla que maneja el encabezado de los documentos, de esta se derivan dos tablas *tinvs_comprobantes_productos*, y *tinvs_comprobante_servicios* que básicamente tienen un funcionamiento similar a lo que se trató en el modelo de talleres, en donde en la una tabla se almacena todos los ítems que tienen relación con los servicios mientras que en la otra todo lo que tiene relación con la tabla *tinvs_productos*.

Otra relación importante que maneja Kairos es el de las personas involucradas en un comprobante, por ejemplo si tratara de una factura de vehículos, existiría: el cliente, el vendedor, la aseguradora, la financiera, y cada uno con sus datos de referencia; para solucionar este problema se utiliza la tabla *tinvs_repartos* la misma que se vincula con

tper_personas y con tinv_comprobantes usando los campos peronsa_id, agencia_id, trx_inv_id, y numero.

TINV_COMPROBANTES				
AGENCIA_ID	VARCHAR2 (3 Byte)	NN (PK)	(IX1,IX5)	
LINEA_ID	VARCHAR2 (10 Byte)	NN (FK)	(IX4)	
TRX_INV_ID	VARCHAR2 (3 Byte)	NN (PK)	(IX5,IX8)	
NUMERO	INTEGER	NN (PK)	(IX6)	
COMPROBANTE	VARCHAR2 (20 Byte)			
FACTURA	VARCHAR2 (20 Byte)			
FEC_COMPROBANTE	DATE	NN		
FEC_VENCE	DATE			
FEC_ENTREGA	DATE			
RETENIDO	VARCHAR2 (1 Byte)			
MONTO_GRA	NUMBER (18,4)	NN		
MONTO_EXE	NUMBER (18,4)	NN		
RECARGO_GRA	NUMBER (18,4)	NN		
RECARGO_EXE	NUMBER (18,4)	NN		
COMISION_PORCENTAJE	NUMBER (5,2)	NN		
DESCUENTO_PORCENTAJE	NUMBER (11,8)	NN		
DESCUENTO_GRA	NUMBER (18,4)	NN		
DESCUENTO_EXE	NUMBER (18,4)	NN		
IVA_PORCENTAJE	NUMBER (5,2)	NN		
IVA_MONTO	NUMBER (18,4)	NN		
IVA_BASE	NUMBER (18,4)	NN		
IVA_ITEMS	VARCHAR2 (1 Byte)	NN		
IVA_CREDITO	VARCHAR2 (1 Byte)			
TOTAL	NUMBER (18,4)	NN		
TIPO_CAMBIO	NUMBER (10,2)			
ANULADO	VARCHAR2 (1 Byte)			
DETALLE	VARCHAR2 (200 Byte)			
AGENCIA_ID_ASIENTO	VARCHAR2 (3 Byte)	(FK)	(IX3)	
NUMERO_ASIENTO	INTEGER	(FK)	(IX3)	
MAYORIZADO	VARCHAR2 (1 Byte)	NN		
KILOMETRAJE	NUMBER (18,4)			
FIN_COMISION	NUMBER (18,4)			
FIN_COMISION_POR	NUMBER (18,4)			
FIN_COMISION_PAGADA	VARCHAR2 (1 Byte)			
USUARIO	VARCHAR2 (50 Byte)			
FEC_CREACION	DATE			
ANULADO_POR	VARCHAR2 (500 Byte)			
TIPO_COMPROBANTE_ID	VARCHAR2 (10 Byte)	(FK)	(IX6)	
FEC_EMITION	DATE			
RETENIBLE	VARCHAR2 (1 Byte)			
MONTO_GRA_M2	NUMBER (22,8)			
MONTO_EXE_M2	NUMBER (22,8)			
RECARGO_GRA_M2	NUMBER (22,8)			
RECARGO_EXE_M2	NUMBER (22,8)			
DESCUENTO_GRA_M2	NUMBER (22,8)			
DESCUENTO_EXE_M2	NUMBER (22,8)			
IVA_MONTO_M2	NUMBER (22,8)			
IVA_BASE_M2	NUMBER (22,8)			
TOTAL_M2	NUMBER (22,8)			
MONEDAD_ID	VARCHAR2 (3 Byte)	(FK)	(IX7)	
TIPO_ID	VARCHAR2 (3 Byte)	(FK)	(IX2)	

Figura 4-9 Tabla tinv_comprobantes (Montalvo, 2019)

En la figura 4-9 se representa la tabla tinv_comprobantes la misma que tiene como claves principales los campos agencia_id varchar de 3, trx_inv_id varchar de 3, este campo almacena el tipo de documento o comprobante con el que se identifica a cara transacción, por ejemplo para facturas de talleres se usa el identificador “SFP” y numero un integer que almacena el número de cada comprobante, además de estos campos se puede observar los campos fec_comprobante, fec_vence, fec_entrega todos de tipo date, que representan las distintas fechas de este proceso. Además existen los campos de valores como total,iva_monto, descuento_gra, todos son de tipo number con precisión 18 y escala de 4, se pueden manejar 4 decimales. Otro campo importante dentro de esta tabla es el campo factura que es un varchar de 20, este campo permite almacenar el número de factura o documento tributario, es decir una factura puede tener como registro en el campo número un dato y en el campo factura otro dato distinto.

Kairos maneja en esta tabla dos números por cada documento el que se imprime o se genera en la factura que tiene validez tributaria es el campo factura, mientras que el campo número es de manejo interno del sistema. Esta tabla maneja 3445139 registros con fecha 18 de diciembre del 2018, aquí se tiene información de PROAUTO C.A. desde el año 2006.

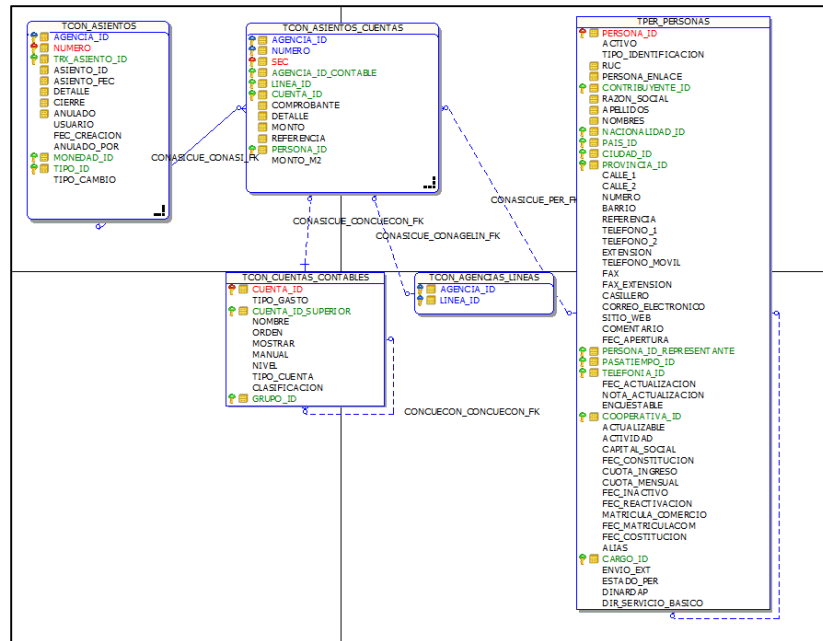


Figura 4-10 Modelo entidad relación Asientos Contables. (Montalvo, 2019)

En la Figura 4-10 se observa el modelo entidad relación con el que Kairos almacena la información de su manejo contable, sin embargo, aquí también se puede encontrar información de las retenciones con una estructura similar a la de comprobantes con la diferencia que al ser retención no tiene manejo ni de servicios ni de inventarios.

En este modelo intervienen las tablas tcon_asientos que es la tabla que maneja el encabezado de los asientos cuya clave principal son los campos agencia_id y numero, las letras “con” hacen referencia a contabilidad. Esta tabla se relaciona con su detalle tcon_asientos_cuentas mediante los campos agencia_id, y numero, y con la tabla tcon_cuentas_contables que es en donde el sistema carga todo el plan de cuentas de la empresa. El campo cuenta_id almacena el número de cuenta, el campo nombre la descripción de la cuenta, también se puede ver una relación con la tabla tcon_agencias_lineas. Esta tabla administra las líneas de negocio por agencia es decir que cuentas de que línea de negocio se afectan dentro de cada agencia, y por último la tabla

tper_personas que ya se ha revisado con anterioridad. Este modelo no es muy extenso en cuanto a la cantidad de campos que se manejan por cada tabla como se ha visto en los otros modelos de este capítulo, sin embargo, es uno de los más importantes porque desde aquí se generan los balances de la compañía.

6. CAPÍTULO V: MIGRACION DE DATA

En este capítulo se revisarán todas las consideraciones que se deben tener al momento de hacer una migración de data, así como también se analizará a fondo el caso de estudio de este trabajo de titulación, el mismo que propone presentar un esquema lógico para migrar data entre Oracle y SQL Server, además de hacer una revisión y análisis de los diferentes escenarios que se podrían presentar al realizar una migración de data, comparando los tipos de data y haciendo referencia al caso de estudio.

6.1. Antecedentes

A mediados del año 2016 Proauto C.A., atravesó un cambio en la administración general. En vista de que las ventas estaban bajando y el gasto superaba los ingresos, ingresa un nuevo Gerente General a la compañía; para ese entonces Proauto C.A. manejaba procesos similares a los que se detallan en capítulos anteriores, pero con el sistema DMS, trabajando en SLQ Server con sus respectivas licencias.

Como parte de los planes de acción de la nueva Gerencia General, surge la necesidad de poder analizar los balances de la compañía de una manera distinta a la que DMS podía ofrecer, con el objetivo de poder tomar decisiones importantes e inmediatas sobre el rumbo de la compañía.

Como experiencia positiva se sugiere hacer el primer acercamiento a la empresa SoftSierra, propietaria del sistema Kairós. Con el objetivo de dar el soporte correspondiente y poder cumplir con este requerimiento. Sin embargo, al tener dos esquemas de modelos de base de datos distintos, no había una certeza de cuánto tiempo podría tomar este desarrollo.

Por esta razón, la Gerencia General dispone migrar la data almacenada por 10 años en SLQ Server a Oracle y utilizar el sistema Kairos. Este sistema en la actualidad funciona en Automotores de la Sierra ya por algunos años, motivo por el cual se tenían excelentes referencias sobre el uso de este y con la ventaja de que la información de los balances estaría en el formato que se necesitaba presentar a la Gerencia.

Una necesidad adicional era tener un mayor control en todas las áreas y mejorar los procesos, con el fin de obtener mayor rentabilidad en cada línea de negocio. Esto implicaba también un cambio tecnológico importante al migrar entre SQL Server a Oracle; siendo Oracle una base de datos con mejores prestaciones en rendimiento y seguridades. Por otro lado, se tenía también la necesidad de poder tener la información de otras empresas del sector y de la marca Chevrolet administradas por los mismos accionistas, bajo un mismo esquema de gestor de base de datos y ERP.

Por todo lo expuesto anteriormente se empieza el proyecto de migrar los datos de una base a otra, es decir de SQL Server a Oracle. Con el fin de poder obtener información de análisis gerencial de manera más clara. Esta directriz se la recibe en octubre del 2016 con un tiempo de entrega 3 meses, es decir se debían migrar 10 años de información y salir a producción el 1ro de enero del 2017.

Sin dimensionar realmente el impacto ni el tiempo real que este proceso llevaría al ser dos estructuras de tablas distintas, con procesos diferentes de almacenamiento de la data en cada base, y con tipos de datos distintos entre sí, el proyecto tuvo una duración de 6 meses aproximadamente. La salida en producción con toda la Data migrada y validada de un DBMS a otro fue en mayo del 2017.

6.2. Secuencia Lógica de Migración

Por los antecedentes de la migración de Data en PROAUTO C.A y debido al corto plazo para terminar el proyecto de migrar información de un Gestor de base de datos a otro, no hubo un periodo de planificación del proyecto. Esta etapa de planificación era indispensable para poder determinar el tiempo, costos, recurso humano, hardware,

licenciamiento, etc. , importantes para iniciar el proyecto y a pesar de ello el proyecto fué realizado sin tener una secuencia lógica de migración de data. Razón por la cual se fueron presentando varios inconvenientes derivados de la no planificación del proyecto.

A continuación, se resumen en 5 FASES el trabajo realizado en su momento en PROAUTO C.A.

Fase 1: Se analizó el hardware con el que se contaba para realizar el trabajo de migración, es decir, el estado de los servidores, software con el que estaban virtualizados, que espacio había disponible en disco para mantener un ambiente de pruebas. Así como también que inversión se necesitaría hacer para temas de licenciamiento; considerando que son dos DBMS distintos y la manera de licenciar cada uno de acuerdo con el hardware, varía por cada fabricante. Por otro lado, de manera macro se revisó la arquitectura de cada modelo entidad relación y las características generales de cada ERP.

Fase 2: Consistió en recopilar la mayor cantidad de información con las diferentes áreas de PROAUTO C.A, con el objetivo de realizar equivalencias de los datos de cómo se deberían visualizar con el otro esquema de base. La primera área con la que se trabajó fue contabilidad. Se elaboró un catálogo de equivalencias del plan de cuentas actual, vs el plan de cuentas al que se migraría y de igual manera se procedió con las líneas de negocio o centros de costos. La siguiente área fue post venta se plantearon catálogos para las equivalencias de la codificación de los modelos del taller agrupados por familias. Por último, con el área de ventas de vehículos de igual manera se generaron las equivalencias de los modelos de vehículos nuevos y se revisó el proceso comercial.

Fase 3: Se preparó el repositorio de datos de Oracle, es decir el hardware y software donde se empezaría a colocar los datos migrados. Para esto se preparó un servidor HP con 16 GB de RAM y discos duros de 600GB, virtualizado con VMWARE ESXi y la máquina virtual con sistema operativo Oracle Linux. Esta es una de las etapas más críticas del proyecto ya que se tuvo que alquilar un servidor, debido a que al momento no se contaba con un equipo con los requerimientos mínimos para garantizar un proceso exitoso. En este servidor virtualizado se instaló Oracle db 11g, y se cargaron las estructuras de tablas que se revisaron en el capítulo 4. Incluyendo además del análisis de cada una de las tablas

del ERP Kairos y del ERP DMS con el fin de poder ir realizando la carga adecuada de los datos históricos.

Fase 4: Se inició con la definición del tipo de migración, por la diferencia en estructuras de ambas bases se decidió utilizar ETL. El proceso operativo de migración es decir la extracción de la data desde la base de datos en SQL Server 2012 hacia Oracle db 11g. Se invirtió un tiempo considerable en analizar la estructura de los datos y tablas.

Fase 5: Fue la puesta en producción del nuevo ERP con toda la data migrada, esta es la fase más importante debido a que en este punto se debe validar la información que fue migrada y realizar el cronograma de puesta en producción en conjunto con todas las áreas de la compañía.

Basado en esta experiencia se plantea un esquema lógico detallado a continuación, incluyendo pasos que complementen las fases mencionadas anteriormente de manera que cuenten con un orden, que permita organizar y garantizar trabajos de migración de data siendo este el aporte de esta disertación.

Esquema Lógico Propuesto de Migración			
FASE	PASO	DESCRIPCION	ESTADO
1	1	Analisis de la Situacion Actual de la Organización	☐
	2	Analisis de la Situacion Futura de la Organización	☐
	3	Determine características del ERP Actual	☐
	4	Determine características del ERP Futuro	☐
2	5	Analisis y Determinacion de DBMS Actual	☐
	6	Analisis y Determinacion DBMS Futuro	☐
	7	Identificacion de Tablas del DBMS Actual	☐
	8	Identificacion de Tablas del DBMS Futuro	☐
3	9	Determinacion de Campos dentro de las Tablas del DBMS Actual	☐
	10	Determinacion de Campos dentro de las Tablas del DMS Futuro	☐
4	11	Definicion del tipo de migración	☐
	12	Ejecucion de proceso de migración	☐
5	13	Pruebas de Data	☐
	14	Puesta en produccion	☐

Figura 5-1 Esquema lógico Propuesto de Migración de DATA (Montalvo, 2019)

La Figura 5-1 representa el esquema lógico que se puede utilizar en un proyecto de migración de datos. El mismo que se presenta a manera de un “checklist” y contiene 14 pasos que están vinculados entre sí en un orden lógico.

PASO 1 y 2: Hacer un análisis general de la situación actual de la empresa u organización en donde se pretende realizar la migración; analizar cuál sería la situación futura de la empresa luego del proceso de cambio o migración que ventajas y desventajas traerá este proyecto. Es importante que se tome el tiempo necesario para estos pasos porque aquí se debe tomar la decisión de si realizar la migración o no. El Capítulo 2 hace referencia a estos pasos.

PASO 3 y 4: Determinar qué características del ERP actual están en uso y que procesos internos se soportan en las funcionalidades del ERP actual, de manera que el impacto que se tenga en los usuarios finales sea mínimo. El objetivo de esto es tener un proceso de migración más limpio, razón por la cual se debe analizar las características del ERP futuro. Todo esto de manera que en la puesta a producción no queden procesos sin responsables ni se pierdan funcionalidades. El objetivo es que se sienta una mejora y una verdadera innovación tecnológica. Como ejemplo se tiene todo el análisis realizado en los capítulos 3 y 4.

PASO 5 y 6: Analizar los dos DBMS el actual y el futuro, en este punto el análisis es una comparación de las características básicas, como son seguridad, velocidad de transaccionalidad, soporte, mantenimiento y por último costos de licenciamiento.

PASO 7 y 8: Se debe realizar la comparación detallada de las tablas más importantes DBMS tanto del actual como del futuro. Analizar sus diagramas de entidad relación, asociar esta información a los procesos con los que opera la organización o empresa. De manera macro hacer las comparaciones de como la data es administrada y almacenada en cada DBMS.

PASO 9 y 10: Aquí se debe realizar un análisis a detalle de los campos más importantes de las tablas que se revisaron en el paso anterior, incluyendo en este trabajo la documentación de los tipos de datos que maneja cada uno de los DBMS. De manera que se puedan tomar decisiones sobre cómo se verá la información luego de ser migrada.

PASO 11 y 12: Una vez realizado cada uno de estos pasos, se puede determinar el tipo de migración que se debería utilizar, además de realizar un cronograma aparte en donde se especifique si se requiere algún tipo de desarrollo adicional para poder ejecutar la migración.

PASOS 13 y 14: En estos pasos se realizan las pruebas de la data migrada y se organiza el paso a producción. Una opción de pruebas de data puede ser con la generación con reportes que permitan medir y comparar la información almacenada en ambos DBMS. Una vez que la información este certificada por las diferentes gerencias de línea y la alta gerencia entonces se procede a certificar las pruebas de data. El siguiente paso es organizar un cronograma con el equipo de apoyo para paso a producción.

6.2.1 Migración de data en Caso de estudio sin aplicar secuencia Lógica.

Basado en la migración de Proauto C.A., donde la migración fue realizada por una directriz gerencial y además que no se contó con el tiempo necesario para poder dimensionar el proyecto. Sin embargo, se pudieron obtener algunos datos como, por ejemplo, el tiempo que tomaba en migrar un año de información era aproximadamente de 8 horas, lo que hacían que las jornadas de migración sean extensas.

El proceso consistía en migrar únicamente grupos de información y luego analizar la consistencia de la data. A continuación, se procedía a realizar las correcciones y ajustes en las consultas SQL con las que se extraía la información de ambos DBMS y por último volver a realizar una nueva la carga de data. Este proceso se realizó en la migración de la información de los documentos tributarios, debido a que para la migración de clientes, la data pasaba con equivalencias de campos en cada tabla como se revisará más adelante. El mismo caso se tuvo para los datos contables, se generó una tabla de equivalencias entre cuentas. Esto simplifico mucho la migración de la contabilidad de la compañía.

Otro de los puntos que tuvieron varios reprocesos al momento de la migración es el manejo del inventario y los datos de talleres puesto que cada sistema maneja de manera distinta el control y el almacenamiento de esta información en sus estructuras de bases de datos. Por esta razón para el proceso final de migración se prepararon 22 equipos que debían ejecutar la migración de la data en paralelo, cada uno procesaba un semestre de cada año arrancando en junio del 2006 hasta marzo del 2017, este proceso tomo alrededor de 20 horas.

6.3. Análisis de DATA

El análisis de data dentro de una migración de distintos DBMS es un pilar fundamental para el éxito de esta. Para esto se necesita revisar un grupo de tablas más significativas del DBMS actual versus el DBMS futuro. Para la migración de PROAUTO C.A. están tablas como la de clientes, la tabla de productos, incluido repuesto y vehículos, y la tabla de documentos. Al mismo tiempo se analiza los tipos de datos que cada tabla tiene y que

comportamiento toma la data al ser transferida entre los dos DBMS, para este caso de estudio se trata de SQL Server 2012 y Oracle db11g.

TPER_PERSONAS				terceros		
	Column Name	Data Type	Allow Nulls	Column Name	Data Type	Allow Nulls
PERSONA_ID	VARCHAR2 (20 Byte)	NN (PK) (IX1)		nit	Tipo_NIT:decimal...	
ACTIVO	VARCHAR2 (1 Byte)			digito	tinyint	
TIPO_IDENTIFICACION	VARCHAR2 (1 Byte)	NN		nombres	varchar(250)	
RUC	VARCHAR2 (13 Byte)	NN (IX5)		direccion	varchar(150)	
PERSONA_ENLACE	VARCHAR2 (10 Byte)	NN (FK) (IX6)		ciudad	varchar(20)	
CONTRIBUYENTE_ID	VARCHAR2 (3 Byte)	NN (FK) (IX2)		telefono_1	varchar(15)	
RAZON_SOCIAL	VARCHAR2 (200 Byte)	NN (IX3)		telefono_2	varchar(15)	
PELLIDOS	VARCHAR2 (50 Byte)	NN (IX4)		fax	varchar(15)	
NOMBRES	VARCHAR2 (50 Byte)	NN (IX5)		apartado_aereo	varchar(10)	
NACIONALIDAD_ID	VARCHAR2 (3 Byte)	NN (FK) (IX7)		tipo_identificacion	char(1)	
PAIS_ID	VARCHAR2 (4 Byte)	NN (FK) (IX3)		pais	varchar(20)	
CIUDAD_ID	VARCHAR2 (4 Byte)	NN (FK) (IX3)		gran_contribuyente	bit	
PROVINCIA_ID	VARCHAR2 (4 Byte)	NN (FK) (IX3)		autoretenedor	bit	
CALLE_1	VARCHAR2 (60 Byte)	NN		bloqueo	tinyint	
CALLE_2	VARCHAR2 (10 Byte)	NN		notas	varchar(250)	
NUMERO	VARCHAR2 (10 Byte)	NN		lista	tinyint	
BARRIO	VARCHAR2 (60 Byte)	NN		concepto_1	varchar(5)	
REFERENCIA	VARCHAR2 (200 Byte)	NN		concepto_2	varchar(5)	
TELEFONO_1	VARCHAR2 (15 Byte)	NN		concepto_3	varchar(5)	
TELEFONO_2	VARCHAR2 (15 Byte)	NN		concepto_4	varchar(5)	
EXTENSION	VARCHAR2 (10 Byte)	NN		concepto_5	varchar(5)	
TELEFONO_MOVIL	VARCHAR2 (15 Byte)	NN		concepto_6	varchar(5)	
FAX	VARCHAR2 (10 Byte)	NN		concepto_7	varchar(5)	
FAX_EXTENSION	VARCHAR2 (10 Byte)	NN		concepto_8	varchar(5)	
CASILLERO	VARCHAR2 (100 Byte)	NN		concepto_9	varchar(5)	
CORREO_ELECTRONICO	VARCHAR2 (200 Byte)	NN		concepto_10	varchar(5)	
SITIO_WEB	VARCHAR2 (200 Byte)	NN		mail	varchar(50)	
COMENTARIO	VARCHAR2 (400 Byte)	NN		pos_num	tinyint	
FEC_APERTURA	DATE	NN		regimen	char(1)	
PERSONA_ID_REPRESENTANTE	VARCHAR2 (20 Byte)	NN (FK) (IX4)		cupo_credito	money	
PASATEMPO_ID	VARCHAR2 (5 Byte)	NN (FK) (IX9)		nit_real	Tipo_NIT:decimal...	
TELEFONO_ID	VARCHAR2 (10 Byte)	NN (FK) (IX1)		condicion	varchar(4)	
FEC_ACTUALIZACION	DATE	NN		vendedor	Tipo_NIT:decimal...	
NOTA_ACTUALIZACION	VARCHAR2 (200 Byte)	NN		fletes	real	
ENCUESTABLE	VARCHAR2 (1 Byte)	NN		es_excento_iva	varchar(1)	
COOPERATIVA_ID	VARCHAR2 (10 Byte)	NN (FK) (IX11)		contacto_1	varchar(50)	
ACTUALIZABLE	VARCHAR2 (1 Byte)	NN		contacto_2	varchar(50)	
ACTIVIDAD	VARCHAR2 (200 Byte)	NN		formato_factura	varchar(10)	
CAPITAL_SOCIAL	NUMBER (18,4)	NN		fecha_creacion	datetime	
FEC_CONSTITUCION	DATE	NN		formato_copias	tinyint	
CUOTA_INGRESO	NUMBER (12,2)	NN				
CUOTA_MENSUAL	NUMBER (12,2)	NN				
FEC_INACTIVO	DATE	NN				
FEC_REACTIVACION	DATE	NN				
MATRICULA_COMERCIO	VARCHAR2 (20 Byte)	NN				
FEC_MATRICULACION	DATE	NN				
FEC_COSTITUCION	DATE	NN				
ALIAS	VARCHAR2 (100 Byte)	NN				
CARGO_ID	VARCHAR2 (3 Byte)	NN (FK) (IX6)				
ENVIO_EXT	VARCHAR2 (1 Byte)	NN				
ESTADO_PER	VARCHAR2 (1 Byte)	NN				
DINARDAP	VARCHAR2 (1 Byte)	NN				
DIR_SERVICIO_BASICO	VARCHAR2 (200 Byte)	NN				

Figura 5-2 Comparación Tabla tper_personas vs. terceros. (Montalvo, 2019)

En la Figura 5-2 se puede visualizar las tablas terceros y tper_personas la primera pertenece a la base de datos de Oracle, y la segunda tabla pertenece a SQL Server 2012. Cada una de estas tablas ya fueron analizadas en los capítulos 3 y 4 respectivamente, estas almacenan los datos de clientes, proveedores, empleados, etc., es decir toda persona natural o jurídica que tiene relación con la empresa. Sin embargo, manejan una estructura de campos distinta. Son unas de las tablas más importantes que fueron migradas.

Si se analiza la clave principal de cada una se puede encontrar las primeras diferencias, para la tabla tper_personas su identificador es el campo persona_id de tipo varchar2 de 20 mientras que en la tabla terceros el identificador es el campo nit que es de tipo decimal(18,0) es decir no recibe decimales y es de precisión 18. En este caso lo que se optó por realizar es convertir el campo nit a un string antes de ser insertado en la nueva tabla. Sin embargo, se deben tener algunas consideraciones por ejemplo DMS almacena

la información del número de identificación con 10 dígitos de esta manera: 1716795719, mientras que Kairos almacena todos los números de identificación con 13 dígitos, es decir, antes de hacer la inserción primero se convirtió el campo a texto y luego se agregaron los numero 001 al final quedando la identificación en Karios de esta manera: 1716795719001. La segunda consideración que se tenía para este tipo de data es que en Ecuador existen números de identificación que empiezan con “0”, por ejemplo, el número: 0700159700. En DMS se almacena sin el cero “0” inicial de la siguiente manera: 700159700, pero al pasar a Kairos se tuvo que convertir a string luego aumentar un cero “0” a la izquierda y por último el 001 la final quedando almacenado así: 0700159700001.

En resumen, los campos que fueron migrados son número de identificación, nombres, apellidos, dirección, tipo de contribuyente, teléfono, correo electrónico, ciudad, país, provincia, para estos campos no hubo mucha dificultad en cuanto a los tipos de datos.

Campo Kairos	Tipo de Dato	Campo DMS	Tipo de Dato
razon_social	Varchar2(300)	nombres	varchar(250)
Calle_1	Varchar2(60)	direccion	Varchar(150)
Calle_2	Varchar2(60)	direccion	Varchar(150)
numero	Varchar2(10)	direccion	Varchar(150)
Teléfono_1	Varchar2(15)	Teléfono_1	Varchar(15)
Correo_electronico	Varchar2(300)	Mail	Varchar(50)
Ciudad_id	Varchar2(4)	Y_ciudad	Varchar(5)

Tabla 5-1 Comparación atributos tablas tper_personas y terceros. (Montalvo, 2019)

En la tabla 5-1 se observa en las dos primeras columnas los nombres de los campos o atributos de las tablas tper_personas de Kairos, con su respectivo tipo de dato y longitud de caracteres y en las dos siguientes columnas estan los campos equivalentes de la tabla personas. De igual manera con su nombre tipo de dato y longitud, también se puede apreciar que al pasar los datos de la tabla origen (DMS) a (Kairos) no existe perdida de data por que la longitud de los datos de destino es más grande que la de los datos de origen.

TINV_PRODUCTOS				referencias		
	Column Name	Data Type	Allow Nulls	Column Name	Data Type	Allow Nulls
PRODUCTO_ID	VARCHAR2(20 Byte)	NN (FK) (108)		codigo	varchar(20)	-
BODEGA_ID	VARCHAR2(2 Byte)	NN (FK) (108)		descripcion	varchar(80)	-
CATEGORIA_ID	VARCHAR2(20 Byte)	NN (FK) (103)		codigo_oferta	varchar(1)	-
NOMBRE	VARCHAR2(60 Byte)	NN		generico	varchar(10)	-
CALIFICACION	VARCHAR2(1 Byte)	NN		clase	varchar(10)	-
IVA_PORCENTAJE	NUMBER(52)	NN		contable	tinyint	-
COLOR_ID	VARCHAR2(20 Byte)	(FK) (1017)		grupo	varchar(10)	-
PESO_ID	VARCHAR2(20 Byte)	(FK) (1019)		subgrupo	varchar(10)	-
VERSION_ID	VARCHAR2(20 Byte)	(FK) (109)		nit	Tipo_NITdecimal...	-
PERIODO	NUMBER(4)			valor_unitario	money	-
CILINDRAJE	VARCHAR2(20 Byte)	NN		porcentaje_iva	real	-
PRECIO_BLOQUEADO	VARCHAR2(1 Byte)	NN		costo_unitario	money	-
FEC_APERTURA	DATE	NN		maneja_inventario	bit	-
LONGITUD_ID	VARCHAR2(20 Byte)	(FK) (101)		und_1	varchar(4)	-
MARCA_ID	VARCHAR2(20 Byte)	(FK) (102)		can_1	real	-
VOLUMEN_ID	VARCHAR2(20 Byte)	(FK) (1022)		und_2	varchar(4)	-
SERIE	VARCHAR2(20 Byte)			can_2	real	-
MOTOR	VARCHAR2(20 Byte)			und_3	varchar(4)	-
CHASSIS	VARCHAR2(20 Byte)			can_3	real	-
COSTO_FOB	NUMBER(18,4)	NN		und_vta	tinyint	-
COSTO_CIF	NUMBER(18,4)	NN		und_com	tinyint	-
CADUCA	VARCHAR2(1 Byte)	NN		impoconsumo	real	-
CLASE_ID	VARCHAR2(20 Byte)	(FK) (1016)		valor_und1	money	-
MODELO_ID	VARCHAR2(20 Byte)	(FK) (1021)		valor_und2	money	-
PARTIDA_ID	VARCHAR2(15 Byte)	(FK) (1025)		valor_und3	money	-
SUBCLASE_ID	VARCHAR2(20 Byte)	(FK) (1015)		conversion	real	-
TIPO_ID	VARCHAR2(20 Byte)	(FK) (106)		otro_impuesto	real	-
UNIDAD_ID	VARCHAR2(20 Byte)	(FK) (1012)		minimo_iva	real	-
PESO_BRUTO	NUMBER(12,6)			minimo_iva_c	money	-
PESO_NETO	NUMBER(12,6)			precio_marcado	varchar(1)	-
LONGITUD_ALTO	NUMBER(12,4)			factor_compra	real	-
LONGITUD_ANCHO	NUMBER(12,4)			factor_venta_1	real	-
LONGITUD_PROFUNDIDAD	NUMBER(12,4)			factor_venta_2	real	-
VOLUMEN	NUMBER(12,4)			factor_venta_3	real	-
APILAR	NUMBER(8)			factor_venta_4	real	-
CARACTERISTICA	VARCHAR2(1000 Byte)			factor_venta_5	real	-
NOTA_PROMOCION	VARCHAR2(2000 Byte)			factor_venta_6	real	-
NOTA_ESPECIFICACIONES	VARCHAR2(4000 Byte)			factor_venta_7	real	-
NOTA_COMPRAS	VARCHAR2(4000 Byte)			factor_venta_8	real	-
NOTA_VENTAS	VARCHAR2(4000 Byte)			factor_venta_9	real	-
CODIGO_ID	VARCHAR2(20 Byte)	NN (FK) (1023)		factor_venta_10	real	-
COMPARTIMIENTO_ID	VARCHAR2(20 Byte)	NN (FK) (104)		fec_cambio_precio	datetime	-
STATUS_ID	VARCHAR2(1 Byte)	NN (FK) (1011)		costo_anterior	money	-
ANALIZAR_MOV	VARCHAR2(1 Byte)			fec_ultima_entrada	datetime	-
CAP_CARGA	VARCHAR2(4000 Byte)			fec_ultima_salida	datetime	-
CAP_PASAJEROS	VARCHAR2(4000 Byte)					
PRECIO_SUGERIDO	NUMBER(18,4)					
PRODUCTO_BC	VARCHAR2(20 Byte)	(AND_AKL_A K3,115,110)				
PRODUCTO_PR	VARCHAR2(20 Byte)	(AND_AKL_A K6,117,110)				
PROMOCION	VARCHAR2(1 Byte)					
RIN	NUMBER(52)					
MEDIDA_ID	VARCHAR2(20 Byte)	(FK) (1010)				
DISEÑO_ID	VARCHAR2(20 Byte)	(FK) (1019)				
LLANTA_ID	VARCHAR2(20 Byte)	(FK) (1014)				
ESTADO_ID	VARCHAR2(20 Byte)	(FK) (1013)				
LLANTA_TIPO	VARCHAR2(1 Byte)					
PLACA	VARCHAR2(20 Byte)					
ENVIO_EXT	VARCHAR2(1 Byte)					
TEST	VARCHAR2(1 Byte)					
DESE	DATE					
HASTA	DATE					
DESCUENTO_MAXIMO	NUMBER(52)					
PRECIO_ID	VARCHAR2(20 Byte)	(FK) (102)				
COLOR_ID_INTERIOR	VARCHAR2(20 Byte)	(FK) (1014)				
PESO_CARROZADO	NUMBER(18,6)					
NUM_EJES	NUMBER(12)					

Figura 5-3 Comparación Tabla tinv_productos vs. referencias. (Montalvo, 2019)

La figura 5-3 representa la comparación de las tablas tinv_productos y referencias, que pertenecen a Kairos y DMS respectivamente. De la misma manera estas tablas fueron revisadas independientemente en los capítulos 3 y 4 donde se analiza el modelo entidad relación de cada ERP. Estas tablas se encargan de almacenar el inventario de la compañía tanto de repuestos como de vehículos. Se puede ver además que aunque son tablas que administran la misma data se manejan de manera distinta en cuanto a los campos que las componen y los tipos de datos que manejan.

En este análisis se identifica que la clave principal de tinv_productos es el campo *producto_id* que es un *varchar2(20)* mientras que en la tabla referencias la clave principal es el campo que se llama código que también es *varchar(20)*. Por otro lado se tiene una diferencia importante en el largo que tienen los campos: *nombre varchar2(60)* y *descripción varchar(80)* de las tablas tinv_productos y referencias respectivamente, este campo almacena el nombre del producto de inventario, el riesgo que se presentó es que

una descripción quede truncada al pasar los datos, sin embargo al realizar un análisis a la data se encontraron únicamente 7 registros que superaban los 60 caracteres que almacena el campo nombre.

Otra particularidad de estas dos tablas es la manera de cómo se almacenan los datos de vehículos en la tabla *tinvs_productos*. Se puede encontrar los campos *motor varchar2(20)*, *chasis varchar2(20)*, *placa varchar2(10)*, entre otros, mientras que en la tabla *referencias* no se encuentra ninguno de estos datos. La razón es porque DMS maneja una tabla adicional *referencias_imp*, en donde están estos datos con los campos *chasis varchar(50)* *motor varchar(50)* y *placa varchar(10)*. Esto genera una complicación en el proceso de migración debido a que la data no es migrada directo de una tabla a otra, sino que es mediante la creación de una consulta SQL que una las tablas *referencias* con *referencias_imp*.

Column Name	Data Type	Allow Nulls
sw	tinyint	
tipo	varchar(4)	
numero	int	
nit	Tipo_NIT.deci...	
fecha	datetime	
condicion	varchar(4)	
vencimiento	datetime	
valor_total	money	
iva	money	
retencion	money	
retencion_causada	money	
retencion_iva	money	
retencion_ica	money	
descuento_ple	money	
fletes	money	
iva_fletes	money	
costo	money	
vendedor	Tipo_NIT.deci...	
valor_aplicado	money	
anulado	bit	
modelo	varchar(5)	
documento	varchar(20)	
notas	varchar(600)	
usuario	varchar(10)	
pc	varchar(20)	
fecha_hora	datetime	
retencion2	money	
retencion3	money	
bodega	smallint	
impoconsumo	money	
descuento2	money	
duracion	smallint	

Column Name	Data Type	Allow Nulls
AGENCIA_ID	VARCHAR2 (3 Byte)	NN (PK) (D6,D8)
LINEA_ID	VARCHAR2 (10 Byte)	NN (FK) (D6)
TRV_INV_ID	VARCHAR2 (3 Byte)	NN (FK) (D6,D8)
NUMERO	INTEGER	NN (PK) (D6)
COMPROBANTE	VARCHAR2 (20 Byte)	
FACTURA	VARCHAR2 (20 Byte)	
FEC_COMPROBANTE	DATE	NN
FEC_VENGE	DATE	
FEC_ENTREGA	DATE	
RETENIDO	VARCHAR2 (1 Byte)	
MONTO_GRA	NUMBER (18,4)	NN
MONTO_EXE	NUMBER (18,4)	NN
RECARGO_GRA	NUMBER (18,4)	NN
RECARGO_EXE	NUMBER (18,4)	NN
COMISION_PORCENTAJE	NUMBER (5,2)	NN
DESCUENTO_PORCENTAJE	NUMBER (11,8)	NN
DESCUENTO_GRA	NUMBER (18,4)	NN
DESCUENTO_EXE	NUMBER (18,4)	NN
IVA_PORCENTAJE	NUMBER (5,2)	NN
IVA_MONTO	NUMBER (18,4)	NN
IVA_BASE	NUMBER (18,4)	NN
IVA_ITEMS	VARCHAR2 (1 Byte)	NN
IVA_CREDITO	VARCHAR2 (1 Byte)	
TOTAL	NUMBER (18,4)	NN
TIPO_CAMBIO	NUMBER (10,2)	
ANULADO	VARCHAR2 (1 Byte)	
DETALLE	VARCHAR2 (200 Byte)	
AGENCIA_ID_ASIENTO	VARCHAR2 (3 Byte)	(FK) (D6)
NUMERO_ASIENTO	INTEGER	(FK) (D6)
MAYORIZADO	VARCHAR2 (1 Byte)	NN
KILOMETRAJE	NUMBER (18,4)	
FIN_COMISION	NUMBER (18,4)	
FIN_COMISION_POR	NUMBER (18,4)	
FIN_COMISION_PAGADA	VARCHAR2 (1 Byte)	
USUARIO	VARCHAR2 (50 Byte)	
FEC_CREACION	DATE	
ANULADO_POR	VARCHAR2 (500 Byte)	
TIPO_COMPROBANTE_ID	VARCHAR2 (10 Byte)	(FK) (D6)
FEC_EMISSION	DATE	
RETENIBLE	VARCHAR2 (1 Byte)	
MONTO_GRA_M2	NUMBER (22,8)	
MONTO_EXE_M2	NUMBER (22,8)	
RECARGO_GRA_M2	NUMBER (22,8)	
RECARGO_EXE_M2	NUMBER (22,8)	
DESCUENTO_GRA_M2	NUMBER (22,8)	
DESCUENTO_EXE_M2	NUMBER (22,8)	
IVA_MONTO_M2	NUMBER (22,8)	
IVA_BASE_M2	NUMBER (22,8)	
TOTAL_M2	NUMBER (22,8)	
MONEDAD_ID	VARCHAR2 (3 Byte)	(FK) (D6)
TIPO_ID	VARCHAR2 (3 Byte)	(FK) (D6)

Figura 5-4 Comparación Tabla *tinvs_comprobantes* vs. *documentos* (Montalvo, 2019)

En la Figura 5-4 se puede observar las tablas *tinvs_comprobantes* y la tabla *documentos* las mismas que forman parte de la estructura de base de datos de Kairos y DMS respectivamente. Estas tablas fueron analizadas previamente y de manera individual en capítulos anteriores. La data que almacenan estas tablas es sobre los documentos que se generan dentro de la compañía estos pueden ser: facturas, notas de crédito, altas de inventario, ingreso de compras, bajas de inventario, transferencias de inventario entre bodegas, guías de remisión, etc.

La clave principal de la tabla *tinvs_comprobantes* está conformada por *agencia_id varchar(3)*, *trx_inv_id varchar2(3)* y *numero* de tipo *integer*. Si se compara con la tabla *documentos* esta tiene únicamente dos campos *tipo varchar(4)* y *numero int*. La data en la tabla *documentos* está conformada por un *tipo* de documento y un *numero* de documento por ejemplo si se trata de una factura de talleres el sistema maneja la codificación “FEPM” que quiere decir Factura Electrónica Post Venta Matriz y el campo *numero* es el número de documento del SRI. Al migrar estos datos a Kairos el campo *tipo* tiene únicamente 3 caracteres se identifican de la siguiente manera “SFP” que significan Salida Factura Post Venta, y el campo *numero* es un secuencial interno.

Para la tabla *tinvs_comprobantes* el campo que almacena el número de documento del SRI es el campo *factura varchar(20)* y la agencia a la que pertenece el documento está identificada por el campo *agencia_id*. Para la migración de estas tablas se necesitó de una tabla de equivalencias debido a que la longitud de estos es diferente, a pesar de que los tipos de datos son compatibles.

En cuanto a la agencia el campo de origen en DMS es el campo *bodega* de tipo *smallint*, entonces para este caso también se necesitó de una tabla de equivalencias además de cambiar el tipo de dato de *smallint* a *varchar(3)*. Por último, para almacenar el campo *factura* en Kairos, se necesitó de un cambio de tipo de dato y también de la unión de una tabla adicional en donde se almacena el establecimiento y el punto de emisión. En resumen, si se tenía por ejemplo en DMS la factura FEPM 001102000023021 la migración quedaría de acuerdo a las tablas 5-2, 5-3, 5-4 que se muestran a continuación:

Tabla Documentos – DMS		
Tipo	Numero	Bodega
FEPM	23021	1030

Tabla 5-2 Extracto de la Tabla documentos DMS (Montalvo, 2019)

Tabla tipo_transacciones - DMS	
Tipo	prefijo
FEPM	001102

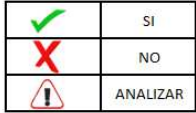
Tabla 5-3 Extracto de la Tabla tipo_transacciones DMS (Montalvo, 2019)

Tabla tinv_comprobantes - Kairos			
Trx_inv_id	Numero	factura	agencia
SFP	47839	001102000023021	001

Tabla 5-4 Extracto tabla tinv_comprobantes Kairos (Montalvo, 2019)

En resumen, para facilidad del personal a cargo de una migración, sería útil tener una tabla de comparación de tipos de datos, que de manera gráfica se puede decidir qué opción tomar cuando el tipo de dato de origen difiere del tipo de dato de destino.

TIPOS DE DATOS		DESTINO									
		TEXTO	NUMBER	DATE	DATETIME	BOOLEAN	MONEY	CLOB	INT	FLOAT	DECIMAL
ORIGEN	TEXTO	✓									
	NUMBER	⚠	✓								
	DATE	⚠	⚠	✓							
	DATETIME	⚠	⚠	✓	✓						
	BOOLEAN	✓	✓	✗	✗	✓					
	MONEY	✗	✓	✗	✗	✗	✓				
	CLOB	✓	✗	✗	✗	✗	✗	✓			
	INT	✓	✓	✗	✗	⚠	⚠	⚠	✓		
	FLOAT	✓	✓	✗	✗	✗	✓	⚠	⚠	✓	
	DECIMAL	✓	✓	✗	✗	✗	✓	⚠	⚠	⚠	✓



✓	SI
✗	NO
⚠	ANALIZAR

Figura 5-5 Análisis de migración por tipos de datos (Montalvo, 2019)

La Figura 5-5 representa un resumen de cómo se puede gestionar la data dentro de un proceso de migración. El lado izquierdo representa de tipos de datos más comunes dentro de ambas estructuras tanto de Kairos como DMS. Este lado representa el origen de la data, mientras que en la parte superior se tiene el destino de igual manera con los mismos tipos de datos.

El objetivo de esta imagen es dar una guía rápida de que se podría hacer al querer pasar data almacena en un tipo de dato a otro, por ejemplo, si la data en el origen es de tipo *clob* y en el destino es *texto*, el paso sería bastante limpio sin embargo para los tipos de datos que almacenan texto se debe siempre tomar en cuenta la longitud de caracteres tanto del origen como el destino. Por otro lado, si el tipo de dato de origen es un *boolean* y el destino es un *date* o *datetime*, en este caso no es posible un paso de dato directo, se debe entonces tomar la decisión de si este dato es importante y analizar posibles opciones.

Por último, si el tipo de dato es una fecha *date* o *datetime* en el origen y en el destino es *texto*, se debe analizar el dato como información y que uso se le va a dar en el destino, es decir hacer un análisis más profundo en cuanto a esta conversión.

6.4. Determinación de ETL

En PROAUTO C.A. la definición del tipo de migración con la que se trabajaría es utilizando ETL. Para esto se contaba con el conocimiento de las estructuras de datos de ambos DBMS. El departamento de sistemas de PROAUTO C.A. tiene la experiencia de trabajar 6 años con DMS y además se tiene el soporte de la empresa SoftSierra quienes desarrollaron Kairos y tienen todo el conocimiento sobre la estructura que maneja su ERP.

En esta etapa se debía generar las sentencias y los procedimientos encargados de leer la Data de SQL server 2012 y luego insertarla en Oracle 11g db, haciendo las equivalencias que se analizaron en el punto anterior y generando los cambios o conversiones de tipos de datos que fueran necesarios.

En términos generales ETL, es el proceso mediante el cual se extrae información de una fuente, la misma es transformada para luego ser cargada en un repositorio distinto del origen. Este proceso no solo puede ser utilizado en migración de data, sino también como una herramienta de análisis de data, con el objetivo de facilitar la generación de reportes, y con la ventaja de que no requiere en muchos casos de tener habilidades técnicas para escribir código. Lo que mejora considerablemente la productividad y el análisis de data en los profesionales.

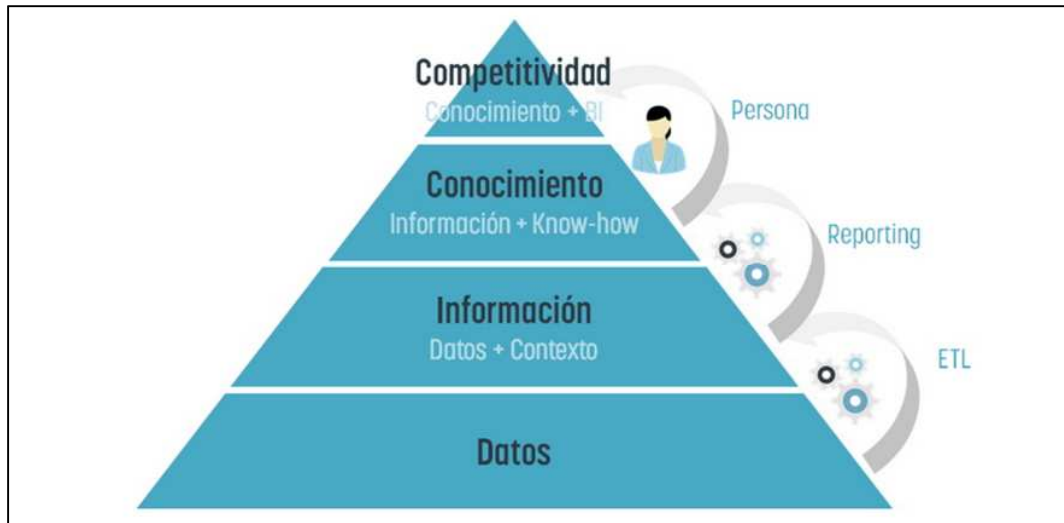


Figura 5-6 Jerarquía de Data con Información y Conocimiento (Valero, 2016)

En la Figura 5-6 se puede ver en como la data es la base del conocimiento la misma que es transformada mediante procesos ETL. Esta data cobra un valor útil luego de pasar este proceso y bajo un contexto específico pasa a ser información. Esta información presentada en un reporte más la experiencia del usuario se convierten en conocimiento, y por último este conocimiento apoyado en herramientas de inteligencia de negocios generan competitividad, y apoyan a la toma de decisiones en las organizaciones. Por esta razón es importante que al momento de la generación de scripts para ETL se tenga claro cuál va a ser el fin de esta data.

Se tomará nuevamente el ejemplo de la cédula de identidad que se analizó en el punto anterior. La necesidad en términos generales era transformar el dato de número a texto,

para esto se necesitaría programar una función que haga esto. Una opción de código sería de la siguiente manera:

```
1 Funcion cedula_num_char
2 Parametros numero_cedula tipo numero
3 Variables cedula_texto tipo texto
4 Inicio
5     cedula_texto = convertir_texto(numero_cedula)
6     Si largo(cedula_texto)=9 o largo(cedula_texto)=12 entonces
7         cedula_texto = concatenar "0" + cedula_texto
8     Fin Si
9     Si largo(cedula_texto)= 10 entonces
10        cedula_texto = cedula_texto + "001"
11    Fin Si
12    Devolver cedula_texto
13 Fin
14
```

Figura 5-7 Función para convertir cedula de número a texto. (Montalvo, 2019)

En la figura 5-7 se puede ver el pseudocódigo propuesto para convertir el campo de cedula de un tipo de dato número a un tipo de dato texto. Se debe tener una consideración adicional que en Karios se almacenan las cedulas de personas o empresas con el “001” al final, como se ve en la línea 10 en donde se concatena el campo *cedula_texto* ya transformado a texto con el “001”.

Como complemento a este ETL, se debería incluir el tamaño del nuevo campo en cuanto a números de caracteres para que no exista un desbordamiento de datos y además eliminar espacios en blanco al inicio y al final de la conversión. En el caso de estudio de PROAUTO C.A, la migración de clientes fue una de las más simples en cuanto a conversión de data puesto que se tenían tablas con estructuras similares, pero con tipos de datos distintos.

Para la migración de cuentas contables, y documentos como facturas, o la relación que cada DBMS maneja entre agencias y bodegas, misma que fue revisada en el análisis de DATA, se podría plantear el siguiente algoritmo de manera genérica. Para todos estos casos seria de la siguiente manera:

```

1  Funcion genérica de equivalencias
2  Parametros bodega texto(4) longitud 4
3  Variable agencia texto(3) longitud 3
4  Inicio
5      Segun bodega Hacer
6          Caso bodega = "1030"
7              agencia = "001"
8          Caso bodega = "3030"
9              agencia = "005"
10         Caso contrario bodega = x
11             agencia = y
12     Fin Segun
13     Devolver agencia
14 Fin

```

Figura 5-8 Función para resolver equivalencias. (Montalvo, 2019)

En la Figura 5-8 representa una posible solución a poder generar equivalencias entre tipos de datos similares, pero de longitudes distintas. Esto se puede aplicar no solo para este caso sino también se podría solucionar con tipos de datos distintos con la particularidad de aumentar una conversión de datos en la variable *agencia*.

Este es un ejemplo que hace referencia a como se almacenan los datos en cada ERP, como se explicó anteriormente para DMS la representación de data de las agencias es almacenado con el campo *bodega* que tiene una longitud de 4 caracteres. Como se puede ver, en la línea 6 la *bodega* "1030" hace referencia a la Matriz. Mientras que en la línea 7 donde está la equivalencia de Kairos con el campo *agencia*, se almacena la información con el dato "001", también para Matriz, es decir, Kairos se apega al manejo de agencias como se las determina en el SRI.

6.5. Pruebas de calidad de Datos

Uno de los puntos más importantes dentro de un proceso de migración es garantizar que al final del proyecto, la información este completa, sea consistente y sobre todo sea un reflejo de su origen. Tomando esto en cuenta se debe siempre considerar realizar pruebas de calidad de los datos, aunque esta tarea en el caso de estudio de PROAUTO C.A., fue realizada durante el proceso de migración por segmentos. Esto no era una garantía que al migrar el 100% de la data esta quede consistente.

Por lo expuesto anteriormente fue necesario antes de la salida a producción realizar una migración completa, con el objetivo de poder validar que toda la información este correcta, completa y sobre todo consistente de acuerdo con lo que se tenía en el ERP anterior. Para este trabajo fue necesario organizar grupos de control, con el líder de cada área de la empresa como son: comercial, talleres, repuestos, contabilidad y cartera.

Las actividades asignadas a cada grupo eran en primer lugar generar reportes en el ERP anterior y compararlos con el equivalente del nuevo ERP. Se generaron balances de meses cerrados, además de saldos de cuentas de ventas y gastos, sin encontrar diferencias, con esto se validó y aprobó la migración contable. De igual manera se procedió para el inventario de vehículos. Se generó el reporte de stock de vehículos en ambos sistemas, encontrando que el número de unidades por modelo también estaba correcto, con esto el área comercial también dio por aprobada la migración de su inventario.

De la misma manera se procedió con el inventario de repuestos y accesorios, con la particularidad que al tener un aproximado de más de 65.000 ítems la validación uno a uno iba a ser casi imposible, por esta razón, se realizó un muestro de los ítems encontrando que el stock estaba igualado en 98%. Para este 2% restante se decidió luego de la migración realizar un inventario físico de toda la compañía con esto se aseguró que el inventario este también validado al 100%.

Por último, el proceso de cuadro de cartera de clientes y proveedores, no tuvo el mismo éxito que las áreas anteriores. Porque ambos sistemas manejan de manera distinta el cruce de valores, es decir las obligaciones con sus respectivos pagos. Además, por el uso de decimales en los valores, no se pudo validar esta información en su totalidad. Para este proceso se tuvo que realizar un trabajo a detalle por parte del área de cartera. Luego de unos 3 meses la cartera, también quedo corregida y validada al 100%.

Un proceso de migración que no siga un esquema lógico podría tomar más tiempo del que inicialmente se pueda plantear. Como se ha revisado a lo largo de este trabajo existen

muchos tipos de migración de data. Razón por la cual es responsabilidad de todas las áreas de una organización involucrarse en estos procesos de manera activa.

Por otro lado, es muy importante tener la apertura de la alta gerencia en cuanto a poder comprender que alcance se quiere dar al proyecto, con el fin de que se pueda garantizar que la data que va a ser trasladada de un DBMS a otro mantenga su confiabilidad y consistencia.

7. CAPÍTULO VI: CONCLUSIONES Y RECOMENDACIONES

7.1. Conclusiones

Se concluye que si se utiliza un esquema lógico de migración los tiempos de un proyecto de migración de data podrían reducirse de manera considerable. Es importante darle el tiempo necesario a cada paso de este esquema y respetar la secuencia. Cada etapa debe ser completada y validada, con el fin de que en la salida a producción se pueda garantizar una buena operatividad en todas las áreas.

Si se decide utilizar ETL como tipo de migración; se debe asignar un tiempo importante en el desarrollo de los scripts, realizar un análisis de los tipos de datos más usados dentro de la compañía, elaborar el respectivo cuadro de análisis de migración por tipo de dato; lo cual facilite el análisis. Además, es necesario que luego de migrada la data se realicen todas las pruebas que se consideren necesarias para que se mantenga la consistencia de la data.

La data almacenada en un gestor de base de datos no siempre tendrá la misma estructura de datos. Existen diferentes tipos de datos entre los diferentes gestores de base de datos, y cada ERP puede almacenar o gestionar sus procesos de manera distinta. También cabe mencionar que en la actualidad el uso de un ERP dentro de compañías medianas, y grandes es indispensable, por la cantidad de información que estas manejan y producen diariamente por todos sus usuarios. Además, que la necesidad de tener reportería gerencial oportuna es cada vez más indispensable.

Se concluye que luego de la migración de data en una empresa los macroprocesos de esta no varían, simplemente se adoptan a las nuevas funcionalidades que pueda tener el nuevo ERP, es importante mencionar que muchas veces las decisiones de migración no vienen del área de TI, sino son directrices de la alta gerencia basadas en la visión que cada gerente pueda tener del mercado en un determinado momento.

Se concluye que no siempre las áreas de TI de las empresas cuentan con todo el diagrama entidad relación de un ERP. Que mientras más complejos son los procesos que este administra más extenso podría ser este diagrama en cuanto a tablas procedimientos, funciones, vistas, etc., independientemente de que DBMS los administre. Por otro lado, el tiempo de vida de un ERP dentro de una empresa no siempre está determinado por la tecnología en el que está desarrollado, sino por las prestaciones que este pueda tener. También por las características de su DBM, o los costos de licenciamiento, también por la flexibilidad de su estructura de base de datos de manera que pueda irse integrando con aplicaciones o módulos satélites que se soporten en el mismo DBMS.

El hardware con el que un DBMS trabaja de manera óptima, no siempre es compatible con otro DBMS ni puede garantizar el mismo performance; cuando se tiene un esquema de virtualización con SQL Server este no será el ideal para que Oracle 11g funcione de manera óptima. Además, el licenciamiento con VMware en SQL Server tiene un costo mientras que Oracle exige utilizar OVM como herramienta de virtualización con el objetivo de no encarecer más sus licencias.

7.2. Recomendaciones

Se recomienda que, si una empresa requiere adquirir o cambiar de gestor de base de datos, se realice un análisis completo y comparativo de todas las características que tiene cada gestor. Como por ejemplo funcionalidades de seguridad, facilidad de instalación y mantenimiento, sistemas operativos compatibles, costos de licenciamiento, etc. Además, se debe revisar si se cuenta o no con personal experto en el mercado para soporte y el

costo que tiene la administración de cada gestor. De tal manera que al momento de tomar la decisión la gerencia general tenga todas las variables claras y opte por la mejor opción en función de las necesidades de la compañía.

Cuando se instale un ERP en una compañía, se solicite al proveedor que entregue el diccionario de datos y los diagramas de entidad relación de los procesos más importantes que se almacenan en la base de datos. Esto con el objetivo de entender como esta almacenada la data y tener un panorama más claro de que información se puede extraer.

Contar con una capacitación sobre las funcionalidades del ERP, capacitar al personal del área de tecnología sobre la estructura y el modelo entidad relación que tienen este ERP; así como la capacitación a usuarios clave que deberán estar familiarizados con el nuevo sistema y para ello es parte fundamental el definir un cronograma de capacitación adecuado.

Antes de iniciar un proceso de migración realizar un análisis completo de los procesos que manejan las compañías, con el fin de poder determinar las brechas que podrían existir al cambiar de sistema, y que afectación podría tener en la operación diaria de los usuarios finales. También es importante involucrar a usuarios clave para realizar un estudio del nuevo ERP, y de esta manera generar un informe que permita de manera profunda, analizar cada área de la compañía y realizar las respectivas comparaciones entre un ERP y otro.

Entender que visión se pretende dar a la migración, porque si se toma solo una visión contable - financiera, en una empresa comercial, algunos datos de segmentación de clientes, prospección, o datos de segmentación de vehículos, podrían quedar afuera generando dificultades en la operatividad y administración de las áreas comerciales.

Es recomendable que cuando se decide cambiar de tecnología entre dos DBMS se cuente con el soporte de un experto en la nueva plataforma a la que se pretenda migrar; con el

objetivo de poder determinar que arquitectura es la más adecuada y que recomienda el fabricante para que el impacto en los costos de licenciamiento sea mínimo. Es decir, entre SQL Server y Oracle se manejan maneras muy distintas de licenciamiento. Por eso se recomienda, que el proceso de compra con Oracle se lo maneje al final del proyecto y luego de realizar todas las pruebas de rendimiento necesarias, debido a que una vez que se notifica a Oracle la intención de compra de licencias se empieza a tener una presión muy grande por parte de la corporación para que se concrete la compra de las mismas; y con seguridad al cabo de poco tiempo de adquiridas las licencias Oracle solicitará una auditoria de licenciamiento.

Analizar la posibilidad de generar o desarrollar una herramienta que permita realizar procesos de migración de data de manera automática, esta herramienta debería tener la capacidad de conectividad entre los DBMS que se analizaron en este trabajo, sin embargo, también se debe considerar que al migrar entre dos DBMS alimentados por dos ERP distintos, se debe tener un amplio conocimiento de las estructuras de cada uno, finalmente la recomendación general en una migración da data, sería que se utilice el esquema lógico de migración propuesto, se genere la respectiva tabla de análisis de tipos de datos, y lo más importante para mitigar el impacto en los usuarios y en la empresa, se debe involucrar a todos los stakeholders en el proceso de tal manera que sean un apoyo dentro del proyecto y en la estabilización del mismo luego de la salida a producción.

BIBLIOGRAFIA

- Abraham Silberschatz, H. F. (2002). *FUNDAMENTOS DE BASES DE DATOS Cuarta Edición*. Madrid: McGRAW-HILL/INTERAMERICANA DE ESPAÑA, S. A. U.
- AcensWhitepaper. (02 de 02 de 2014). *ACENS*. Obtenido de <https://www.acens.com/wp-content/images/2014/02/bbdd-nosql-wp-acens.pdf>
- Amazon Web Services. (11 de 11 de 2018). Obtenido de https://docs.aws.amazon.com/es_es/amazondynamodb/latest/developerguide/Introduction.html
- Athreya, J. R. (Junio 2007). *Oracle Database 11g: Información General sobre Real Application Testing y su Capacidad de Administración*. Redwood Shores, CA 94065 U.S.A. : Oracle.
- Aula 301. (23 de 10 de 2018). Obtenido de <https://aula301.com/tipos-datos-podemos-utilizar-mongodb/>
- Beynon-Davies, P. (2014). *Sistemas de bases de datos Tercera Edición*. Barcelona: Editorial REVERTÉ S.A.
- Cabello, M. V. (2010). *Introducción a las Bases de Datos Relacionales*. Madrid: Vison Libros.
- Date, C. (2001). *Introducción a los Sistemas de bases de datos 7ma Edición*. Mexico: Pearson Prentice Hall.
- Ecuador, G. M. (2018). *Reporte Ventas Octubre 2018*. Quito: General Montors.
- GenBeta. (31 de 05 de 2011). Obtenido de <https://www.genbeta.com/desarrollo/una-introduccion-a-mongodb>
- Grupo CONSISA. (19 de 04 de 2017). Obtenido de <http://www.consisa.com/index.php/es/para-que-sirve-un-erp-en-mi-empresa/>
- Lucidchart. (10 de 10 de 2018). Obtenido de <https://www.lucidchart.com/pages/es/qu%C3%A9-es-un-diagrama-entidad-relaci%C3%B3n>
- Montalvo, G. (2019). Quito.
- Ochando, P. M. (20 de 03 de 2014). *Automatización de Unidades de Información*. Obtenido de <http://ccdodoc-automatizacion.blogspot.com/2014/03/la-migracion-de-datos-exportacion-e.html>
- Pandorafms Monitoring Blogs. (20 de 04 de 2017). Obtenido de <https://blog.pandorafms.org/es/bases-de-datos-nosql/>
- Power Data. (13 de 11 de 2013). Obtenido de <https://blog.powerdata.es/el-valor-de-la-gestion-de-datos/bid/348870/qu-se-entiende-por-integridad-de-los-datos>

- PowerData*. (10 de 10 de 2018). Obtenido de <https://www.powerdata.es/migracion-de-datos>
- Proauto C.A.* (19 de 10 de 2018). Obtenido de <https://proautochevrolet.com.ec/>
- Rabeller, C. (25 de 07 de 2017). *Microsoft Docs*. Obtenido de <https://docs.microsoft.com/es-es/sql/database-engine/sql-server-database-engine-overview?view=sql-server-2017>
- Rouse, M. (11 de 11 de 2018). *Techtarget*. Obtenido de <https://searchdatacenter.techtarget.com/es/definicion/Sistema-de-gestion-de-bases-de-datos-relacionales-RDBMS>
- Sanchez, J. (2004). *Diseño Conceptual de Bases de Datos*. California : Creative Commons.
- SCRIBD*. (4 de 11 de 2018). Obtenido de <https://es.scribd.com/document/179464172/Oracle-Designer-docx>
- Sistema Manejador de Base de Datos (SMBD)*. (11 de 10 de 2018). Obtenido de <http://sistemamanejadordebasededatossmbd.blogspot.com/2011/02/construir-una-base-de-datos-taller.html>
- Stanek, W. R. (2013). *Microsoft SQL Server 2012 Guia del Administrador*. Madrid: Ediciones Anaya Multimedia.
- Tecnología e Infomática*. (1 de 08 de 2009). Obtenido de <http://latecnologiavirtual.blogspot.com/2009/08/datos.html>
- tic.portal*. (22 de 09 de 2018). Obtenido de <https://www.ticportal.es/temas/enterprise-resource-planning/que-es-sistema-erp>
- Valero, M. (16 de 05 de 2016). *bigeek*. Obtenido de Que es una ETL: <https://blog.bigeek.com/que-es-una-etl/>
- Yera, A. C. (02 de 10 de 2007). *Diseño y Programación de Bases de Datos*. Madrid: Vision Libros. Obtenido de <https://www.marcoteorico.com/curso/31/fundamentos-de-bases-de-datos/405/niveles-de-abstraccion>

ANEXOS

Anexo 1: Organigrama General de Proauto.pdf

Anexo2: Diagrama de Procesos del área Comercial.

Anexo3: Diagrama de Procesos de Talleres.

GLOSARIO

AGLP: La licencia pública general de Affero (en inglés, Affero General Public License, también Affero GPL o AGPL) es una licencia copyleft derivada de la Licencia Pública General de GNU diseñada específicamente para asegurar la cooperación con la comunidad en el caso de software que corra en servidores de red. La Affero GPL es íntegramente una GNU GPL con una cláusula nueva que añade la obligación de distribuir el software si éste se ejecuta para ofrecer servicios a través de una red de ordenadores.

ASCII: (acrónimo inglés de American Standard Code for Information Interchange, Código Estándar Estadounidense para el Intercambio de Información, es un código de caracteres basado en el alfabeto latino, tal como se usa en inglés moderno. Fue creado en 1963 por el Comité Estadounidense de Estándares (ASA, conocido desde 1969 como el Instituto Estadounidense de Estándares Nacionales, o ANSI).

DBMS: Definición de DBMS. (Data Base Management System). Son las siglas en inglés para los Sistemas de Gestión de Bases de Datos (SGBD). Bajo este nombre se conoce a productos de fabricantes como Oracle, Sybase, Informix, Ingres, Borland, Microsoft, IBM, etc.

EBCDIC: (Extended Binary Coded Decimal Interchange Code - Código de intercambio decimal de código binario extendido), es un código estándar de 8 bits usado por computadoras mainframe IBM. IBM adaptó el EBCDIC del código de tarjetas perforadas en los años 1960 y lo promulgó como una táctica customer-control cambiando el código estándar ASCII.

ETL: Extract, Transform and Load («extraer, transformar y cargar», frecuentemente abreviado ETL) es el proceso que permite a las organizaciones mover datos desde múltiples fuentes, reformatearlos 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. Los procesos ETL también se pueden utilizar para la integración con sistemas heredados. Se convirtieron en un concepto popular en los años 1970.

KVM: La máquina virtual K (en inglés K virtual machine, KVM) es una máquina virtual desarrollada por Sun Microsystems (actualmente propiedad de Oracle) a partir de una versión simplificada de la máquina virtual Java. Es la máquina virtual que sirve de base a Java ME (Java Micro Edition), una plataforma enfocada al desarrollo e implantación de aplicaciones Java para pequeños dispositivos como teléfonos móviles, PDA, etc. Sobre la KVM se constituye una serie de clases genéricas, organizadas en dos perfiles según las prestaciones del dispositivo final.