

Capítulo 3 ATEXEM: Análisis Textil

Empresarial

El objetivo de este capítulo es presentar a ATEXEM, un sistema de análisis de la información empresarial de Textiles Carmelita. Este capítulo es organizado de la siguiente manera. La sección 3.1 presenta la arquitectura general del sistema. La sección 3.2 presenta el esquema multidimensional de ATEXEM. Las secciones 3.3 y 3.4 presentan el análisis y el diseño del sistema de las fases de construcción y de explotación. La sección 3.5 concluye el capítulo.

3.1 Arquitectura general

La figura 3.1 muestra la arquitectura de ATEXEM para apoyar la creación y almacenamiento de los datos recuperados de las ventas y los inventarios de “Textiles Carmelita”. La arquitectura de ATEXEM está constituida por dos módulos principales, el de construcción y el de análisis. El módulo de construcción está formado por los extractores. El módulo de análisis está formado por la interfaz gráfica de usuario y el motor de consultas OLAP.

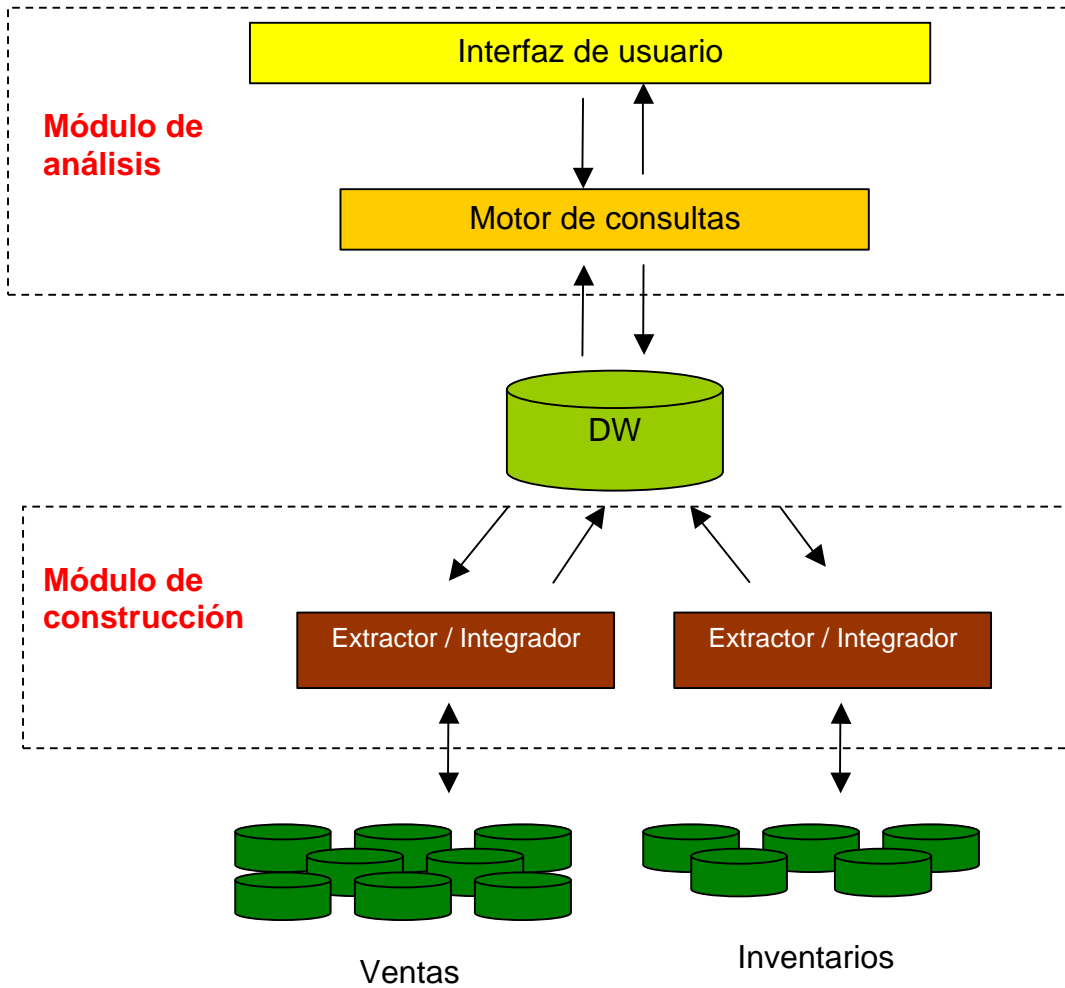


Figura 3.1 Arquitectura de ATEXEM

3.2 Esquemas de ATEXEM

Un *data warehouse* (DW) necesita definir un esquema para organizar los datos almacenados dentro del DW. Para llevar a cabo un diseño correcto es necesario analizar los datos que se ingresarán, así como las necesidades de las aplicaciones (Ver Anexo B).

ATEXEM implementa dos esquemas multidimensionales que son instrumentados a través de un esquema relacional. Estos esquemas de representación se implementaron para nuestros dos cubos multidimensionales el de ventas e inventarios. A continuación describiremos la representación de los esquemas multidimensionales.

3.2.1 Esquema de Ventas

Para construir el cubo de ventas, primero se llevaron a cabo varias entrevistas con la empresa para determinar los requerimientos del sistema (Ver Anexo A). A lo largo de las pláticas se analizó la estructura de las fuentes y su formato. Con base en esto, se decidió que este cubo estaría formado por tres dimensiones:

Producto: grupo de producto correspondiente a un cliente en una fecha.

Clientes: clientes ubicados en diferentes agrupaciones geográficas.

Tiempo: fecha en el tiempo, como día, mes o año.

Estas dimensiones están definidas por niveles de agregación o granos. Los niveles definidos por dimensión se presentan a continuación:

Producto: Tela → Grupo → Familia.

Cliente: Ciudad → Región → Estado → País.

Tiempo: Día → Mes → Año.

País es un nivel que engloba todas las tuplas obtenidas en la dimensión Cliente. Por último definimos como medida aditiva del cubo de ventas los metros vendidos. La figura 3.2 muestra el cubo de ventas definido por las dimensiones Producto, Clientes y Tiempo. Un ejemplo de las medidas almacenadas indica que se vendieron 15.5 metros de tela “Escocés” en “Atoyatempan”.

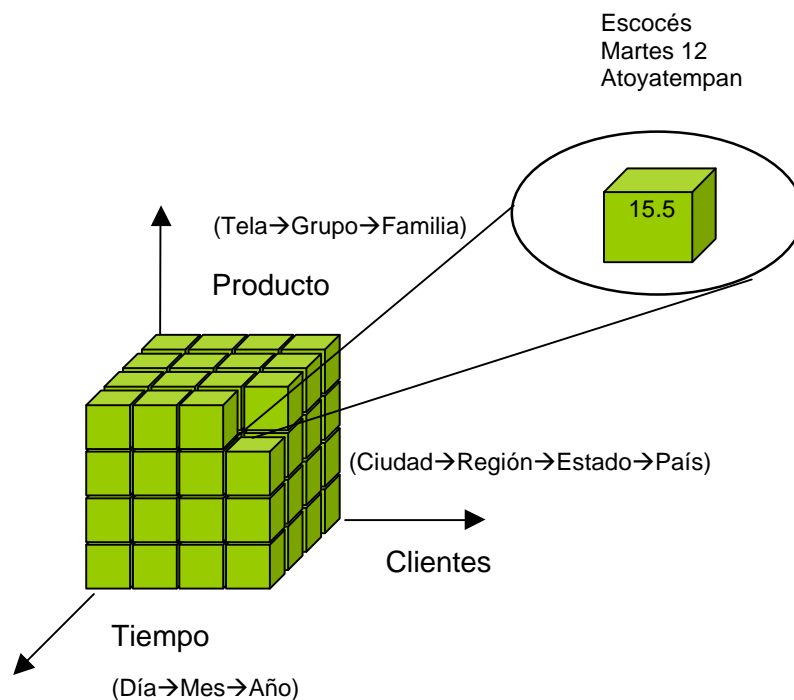


Figura 3.2 Cubo de ventas

Los dominios de cada dimensión se definen a continuación:

Dominios de Productos

Telas: { Escocés, MascotínA, Popelina, Hilo, MascotínP, Franela, Listón, Manta, Toalla, Tira Bordada, Cortina }

Grupos: { Grupo1, Grupo 2 } en donde

Grupo 1: { Escosés, MascotónA, MascotínP, Hilo, Tira Bordada}

Grupo 2: { Popelina, Franela, Listón, Manta, Toalla, Cortina}

Familia: { Familia 1} en donde

Familia 1: { Grupos}

Dentro de este dominio de productos se tienen varias telas. Cada una pertenece a un grupo dependiendo de las características de la tela. Los grupos a su vez forman una familia de productos.

Dominios de Clientes

Ciudades: { Atoyatempan, Atlixco, San Bernabé, Tlanepantla, Los Reyes, Almecatla, Balbuena}

Regiones: { Región Puebla, Región México } en donde

Región Puebla: { Atoyatempan, Atlixco, San Bernabé, Almecatla}

Región México: { Tlanepantla, Los Reyes, Balbuena}

País: { País 1} en donde

País 1: { Regiones}

Dominios de Tiempo

Día: {1 ... 31} según el mes

Mes: {Enero ... Diciembre}

Año: {1998 ... 2004}

El dominio de la dimensión tiempo no se definió ya que se considera trivial la granularidad así como su dominio en días, meses y años. Una vez que contamos con el dominio de las tres dimensiones y con una medida para el cubo de ventas, se obtuvo un esquema en copo de nieve que a continuación se presenta en la figura 3.3.

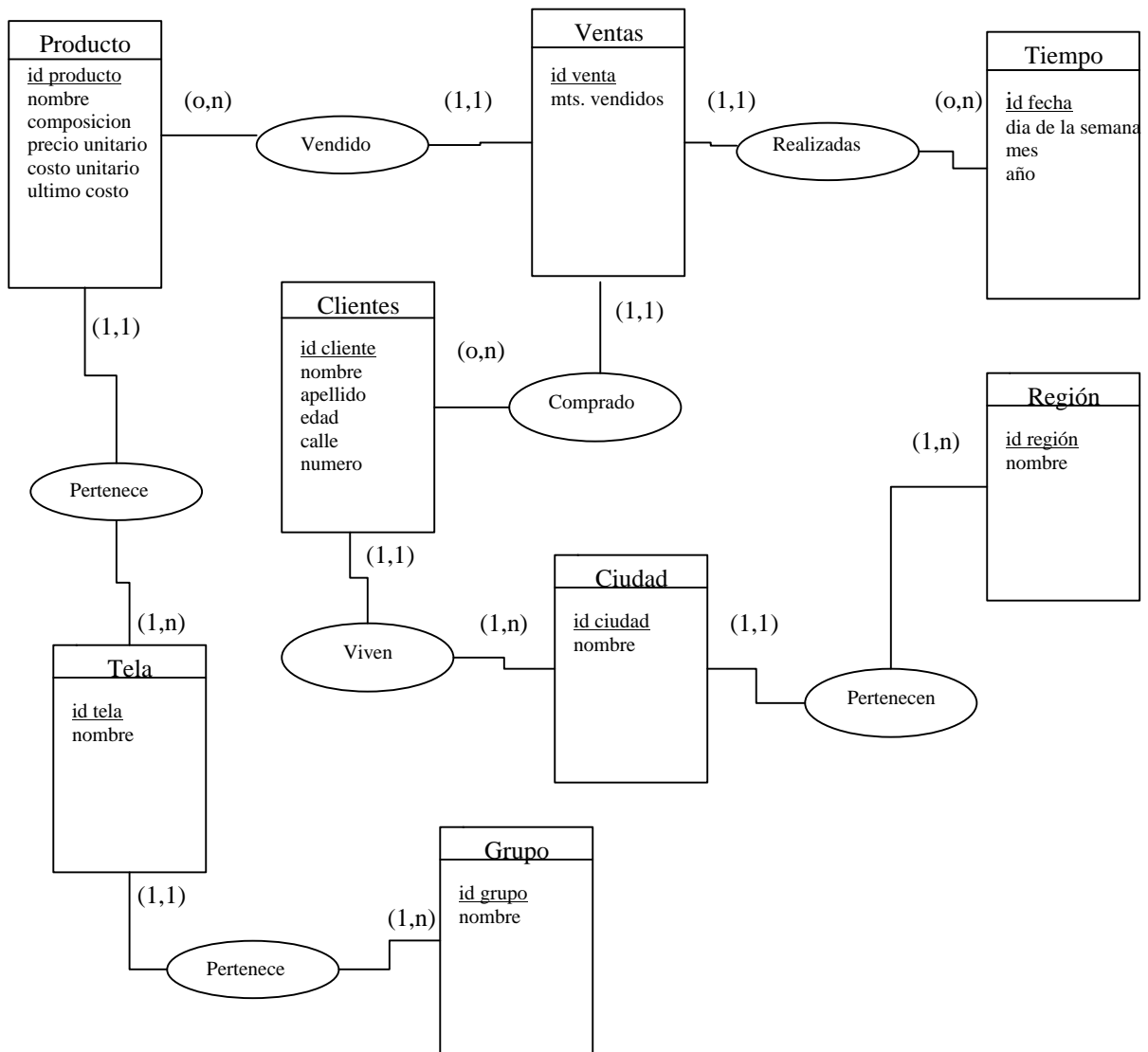


Figura 3.3 Modelo entidad relación del esquema de ventas en copo de nieve

3.2.2 Esquema de inventarios

Para diseñar el cubo de inventarios, primero se procedió a llevar a cabo el análisis de la información que nos proporcionan las fuentes. Basándonos en eso, se decidió que éste cubo estaría formado por tres dimensiones:

Producto: Productos comercializados por la empresa.

Proveedor: Haciendo referencia a los proveedores de los productos antes mencionados.

Tiempo: Como su nombre lo dice, hace referencia a un momento en el tiempo.

Estas dimensiones están integradas por varios niveles de agregación, dando por resultado las siguientes granularidades:

Producto: Tela → Grupo → Familia.

Proveedor: Ciudad → Región → País.

Tiempo: Día → Mes → Año.

Por último como medida se tienen los metros almacenados en el inventario de los productos que se comercializan en la empresa. La figura 3.4 representa el cubo de datos para el almacenamiento de los productos en el DW de inventarios.

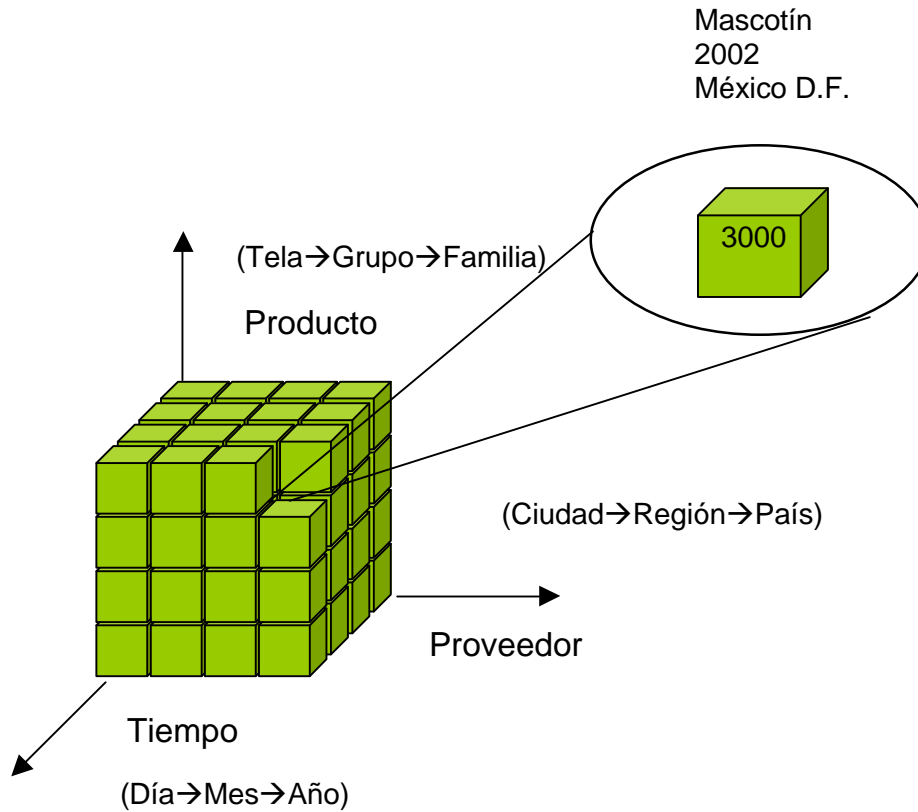


Figura 3.4 Cubo de inventarios

Los dominios de cada dimensión se definen a continuación:

Dominios de Productos

Telas: {Escocés, MascotínA, Popelina, Hilo, MascotínP, Franela, Listón, Manta, Toalla, Tira Bordada, Cortina}

Grupos: {Grupo1, Grupo2} en donde

Grupo 1: { Escocés, MascotónA, MascotínP, Hilo, Tira Bordada}

Grupo 2: { Popelina, Franela, Listón, Manta, Toalla, Cortina}

Familia: { Familia 1} en donde

Familia 1: {Grupos}

Dominios de Proveedor

Ciudades: { Puebla, Tlaxcala, Mexico D.F., Monterrey, Guadalajara}

Regiones: { Región Centro, Región Norte}

Región Centro: { Puebla, Tlaxcala, México D.F.}

Región Norte: { Guadalajara, Monterrey}

País: { País 1}

País 1 : { Regiones}

Dominios de Tiempo

Día: {1 ... 31} según el mes

Mes: {Enero ... Diciembre}

Año: {1998 ... 2004}

Una vez que contamos con el dominio de cada dimensión y la medida del cubo se creó un esquema en copo de nieve que a continuación se muestra en la siguiente figura.

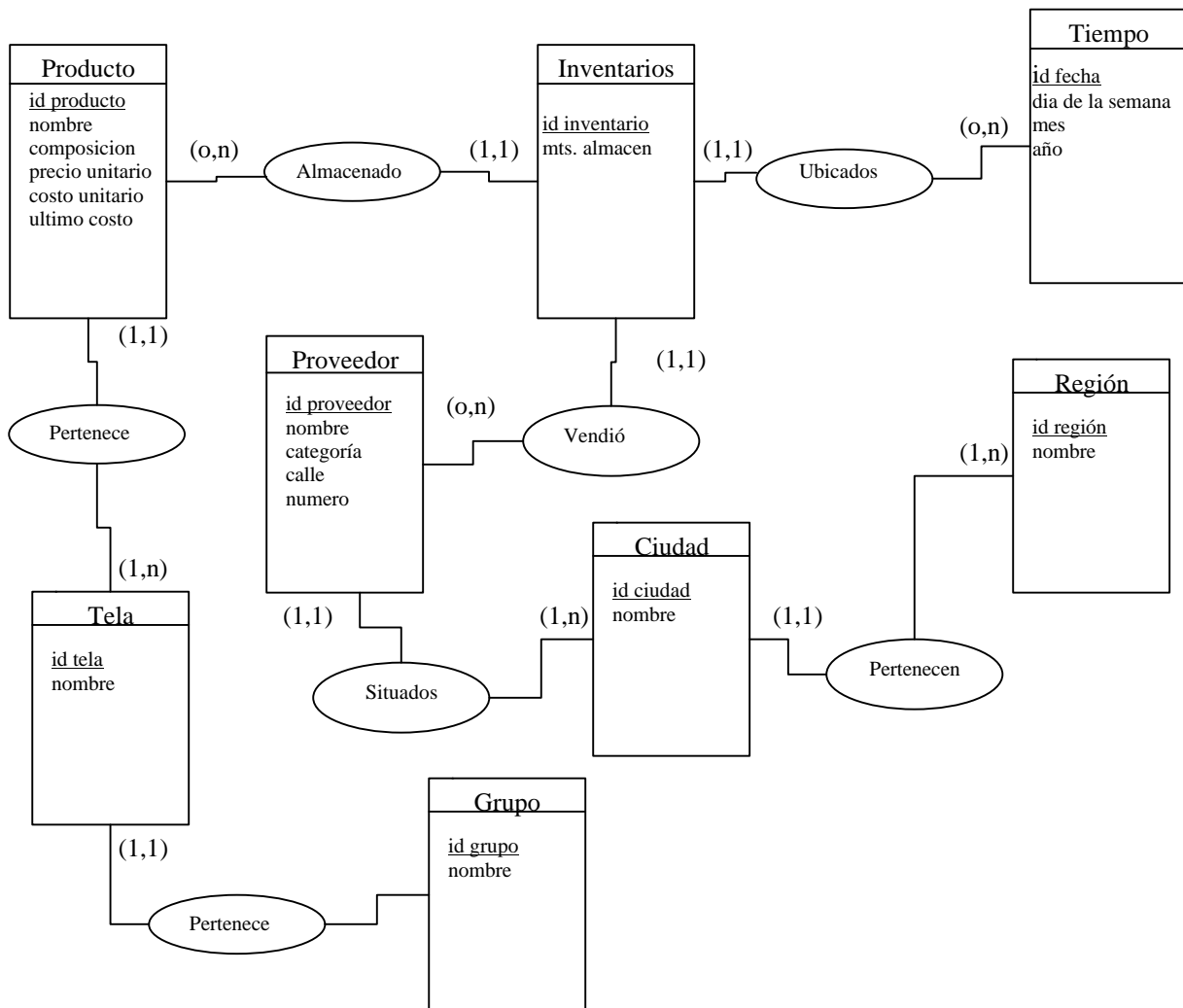


Figura 3.5 Modelo entidad relación del esquema de inventarios en copo de nieve

3.3 Construcción

La figura 3.6 demuestra que cada extractor - integrador sigue la ruta de la localización de la información de las fuentes. La forma en la que extrae la información es particular para cada una de ellas. Ya que como se aprecia en el gráfico los extractores son los encargados de homogenizar, limpiar e integrar la información antes de poblar el DW.

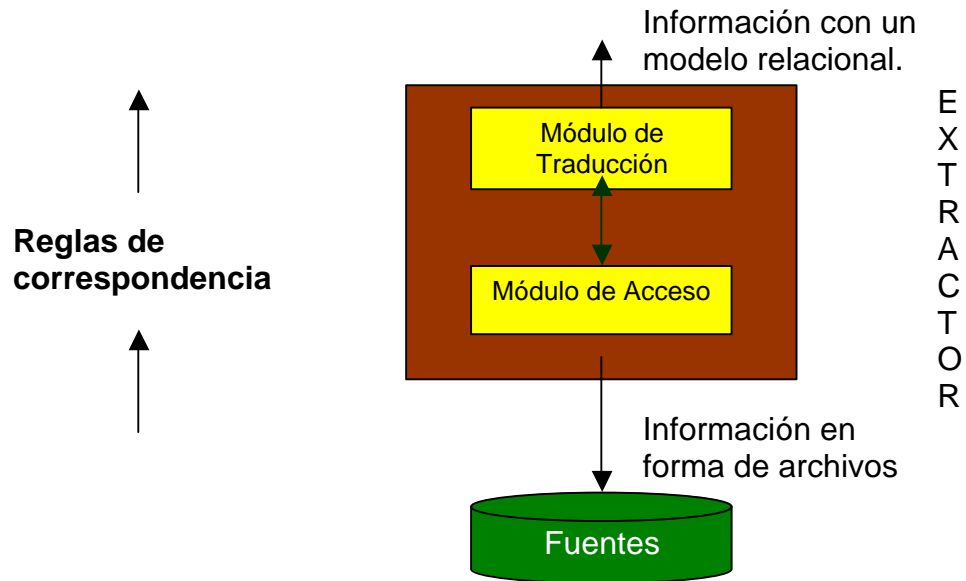


Figura 3.6 Arquitectura de construcción

El proceso de construcción y mantenimiento consiste en extraer los datos de las fuentes, transformarlos de matrices (representadas en hojas Excel) a un modelo relacional, verificarlos, asegurar su integridad y eventualmente refrescar el contenido del DW. A continuación se explican los procesos de extracción y mantenimiento.

3.3.1 Fuentes SAE

SAE es un sistema externo el cual provee las fuentes de ATEXEM. El objetivo de éste sistema dentro de la empresa “Textiles Carmelita” es controlar el ciclo de todas las operaciones de compra-venta de la empresa en forma segura y confiable.

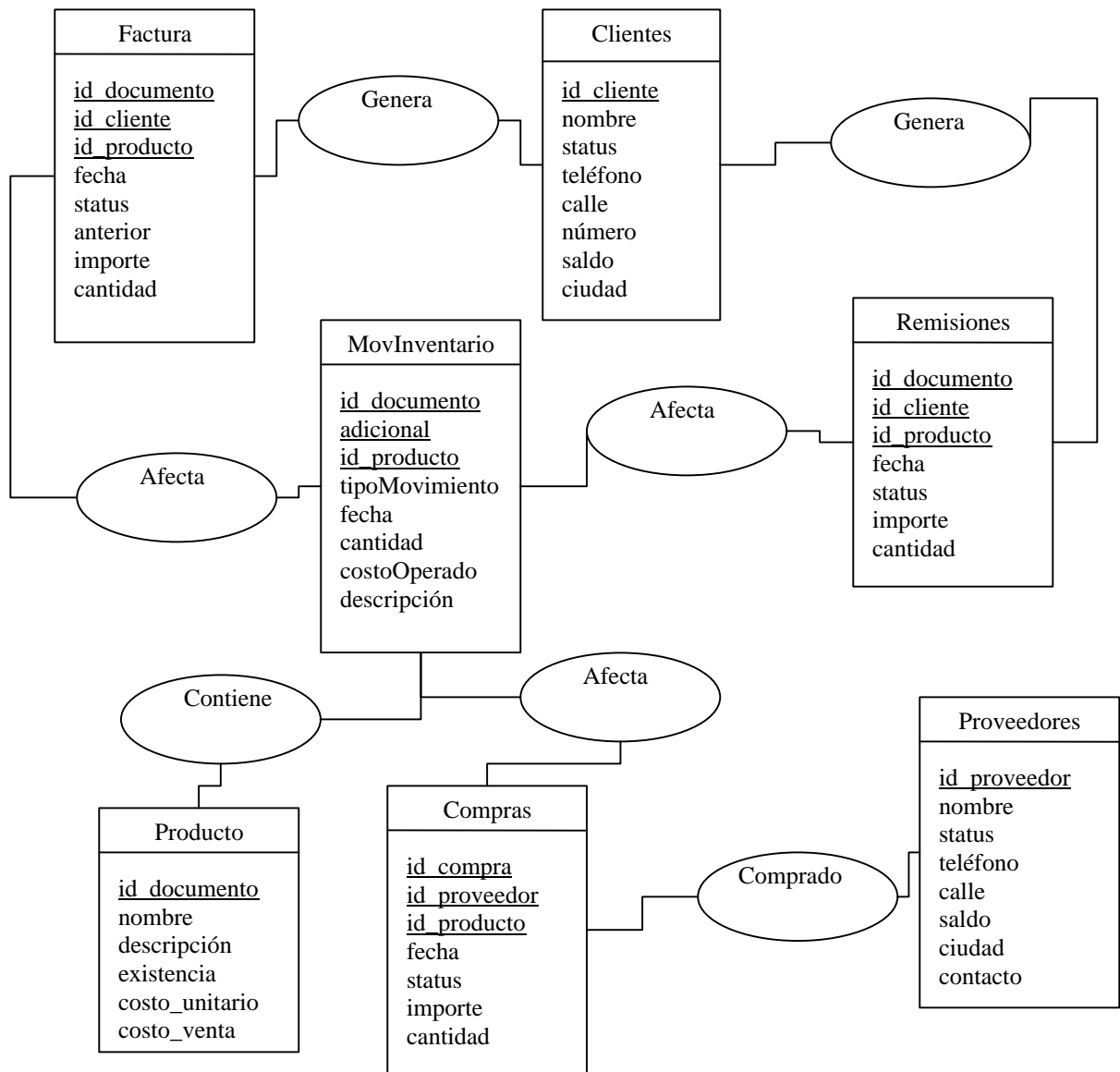


Figura 3.7 Esquema entidad relación de SAE

En la figura 3.7 se muestra el esquema entidad – relación de la fuente SAE. Las entidades de factura y remisiones son generadas por la entidad de clientes y se basan en las ventas realizadas de la empresa. Mientras que la entidad de compras se basa en la adquisición de mercancía que realiza la empresa. Las tres entidades modifican a la entidad movInventario, ésta concentra toda la información de entrada y salida de los productos definidos en la entidad del mismo nombre.

3.3.2 Extracción

“Textiles Carmelita” cuenta con fuentes con una estructura de hojas de cálculo, dichas fuentes se encuentran almacenadas en un sistema llamado SAE. También se detectó que la información carece de limpieza ya que no se encuentra libre de errores.

ATEXEM cuenta con dos extractores. Uno para el cubo de ventas y otro para el cubo de inventarios. Ambos cuentan con características similares, sin embargo cada uno almacena información diferente a pesar de obtenerla del mismo esquema.

En la figura 3.8 se muestra el funcionamiento de construcción de los extractores. Este funcionamiento cuenta con: un módulo que accede las fuentes y un módulo de transformación. El módulo de transformación recibe la información en un formato de EXCEL y lo traduce a un modelo relacional. Para este punto, hemos extraído ya la información de nuestras fuentes. Lo que queda es limpiar esa información y asegurarnos

que va libre de errores. Es decir, eliminar redundancia y validar los datos para finalmente permitir la entrada de la información al DW.

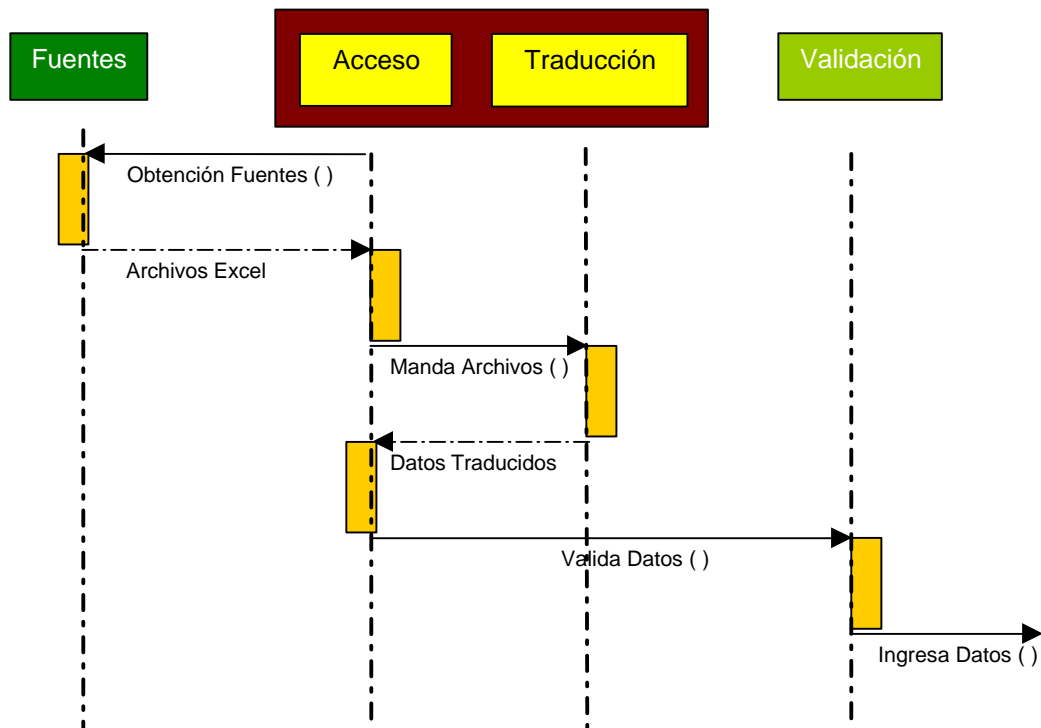


Figura 3.8 Funcionamiento de construcción

Como se muestra en la Figura 3.8, la fuente brinda la información al extractor, posteriormente el extractor recibe los datos en formato de tablas (estructura matricial) y los transforma a un modelo relacional para ser introducidos al repositorio.

Para poder integrar la información se debe transformarla de hojas Excel a un esquema multidimensional. Esto se hace por medio de reglas de correspondencia que a continuación se mencionan:

T1: (Si MovInventario.TipoMovimiento es una venta) entonces

Ventas.id_Cliente = MovInventario.adicional

Ventas.id_Tela = MovInventario.id_producto

Ventas.id_Fecha = MovInventario.fecha

Ventas.mts_Vendidos = MovInventario.cantidad

En la regla T1 se hace la correspondencia de la entidad MovInventario de la fuente con Ventas del esquema multidimensional.

T2: (Si MovInventario.TipoMovimiento es una venta) entonces

Cliente.id_Ciudad = (MovInventario |X| Clientes).ciudad

Cliente.id_Cliente = MovInventario.adicional

Cliente.nombre = (MovInventario |X| Clientes).nombre

Cliente.calle = (MovInventario |X| Clientes).calle

Cliente.número = (MovInventario |X| Clientes).número

Cliente.teléfono = (MovInventario |X| Clientes).teléfono

En la regla T2 se hace la correspondencia de la entidad MovInventario y los Clientes con los Clientes en el esquema multidimensional. Esto si es que el tipo de movimiento es una venta.

T3: (Si MovInventario.TipoMovimiento es una compra) entonces

Inventarios.id_Tela = MovInventario.id_producto

Inventarios.id_Fecha = MovInventario.id_fecha

Inventarios.id_Proveedor = MovInventario.adicional

Inventarios.mts_almacenados = MovInventarios.cantidad

En la regla T3 se hace la correspondencia de la entidad MovInventario de la fuente con Inventarios del esquema multidimensional si es que el tipo de movimiento es una compra.

T4: (Si MovInventario.TipoMovimiento es una compra) entonces

Proveedori.id_Ciudad = (MovInventario |X| Proveedores).id_proveedores

Proveedori.id_proveedor = MovInventario.adicional

Proveedori.nombre = (MovInventario |X| Proveedores).nombre

Proveedori.telefono = (MovInventario |X| Proveedores).telefono

En la regla T4 se hace la correspondencia de la entidad MovInventario y los Proveedores con los Proveedores en el esquema multidimensional. Esto si es que el tipo de movimiento es una compra.

Estas son algunas de las reglas de correspondencia utilizadas sobre los esquemas de la fuente, las demás se encuentran en el Anexo E. En las reglas se hace mención a un campo llamado adicional, éste varía su contenido dependiendo del tipo de movimiento que esté registrado y sus valores pueden ser un id_proveedor para compras o un id_cliente para ventas. Una vez que ya se llevó a cabo éste proceso de correspondencia se deben aplicar reglas similares de validación y limpieza para asegurar la integridad de la información.

3.4 Explotación y análisis

ATEXEM utiliza la tecnología OLAP con sus operadores *slice 'n dice*, *roll-up* y *drill-down* (ver sección 3.5.2). Para diseñar estos operadores se introduce a continuación el concepto de vista.

3.4.1 Vistas

Las vistas se pueden definir como una relación virtual derivada a partir de otras relaciones. Desde el punto de vista conceptual se trata de un cubo que almacena medidas agregadas con respecto a subconjunto de dimensiones cada una en un nivel específico. Se han diseñado dos entidades para administrarlas.

En la primera tabla, llamada condición, se almacenarán las condiciones o los datos que identificarán a esa consulta. En la segunda tabla, llamada consulta, se almacenarán los resultados de la consulta junto con las otras funciones de agregación que sean necesarias. Con la creación de éstas dos tablas se llevará a cabo un diseño eficaz de los operadores OLAP.

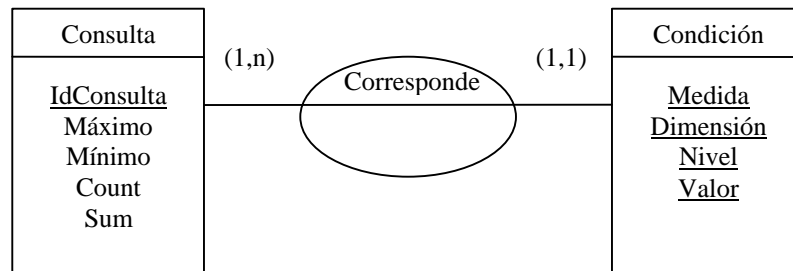


Figura 3.9 Esquema entidad relación de la administración de consultas

En la figura 3.9 se muestran las dos tablas y su relación mediante llaves de referencia en este caso la llave de condición son todos los atributos y de esta manera no se permiten condiciones similares y para cada condición se va a tener una tupla de respuestas en la tabla de consulta.

3.4.2 Operadores

El operador *slice 'n dice* permite restringir los valores asociados a una o varias dimensiones del cubo. Nos permite reducir la información a analizar. Éste operador no es otra cosa que un filtro. Es como si cortáramos con unas cuchillas las cuales nos restringen el cubo para analizar un subconjunto de la información. El diseño del operador es el que se muestra a continuación.

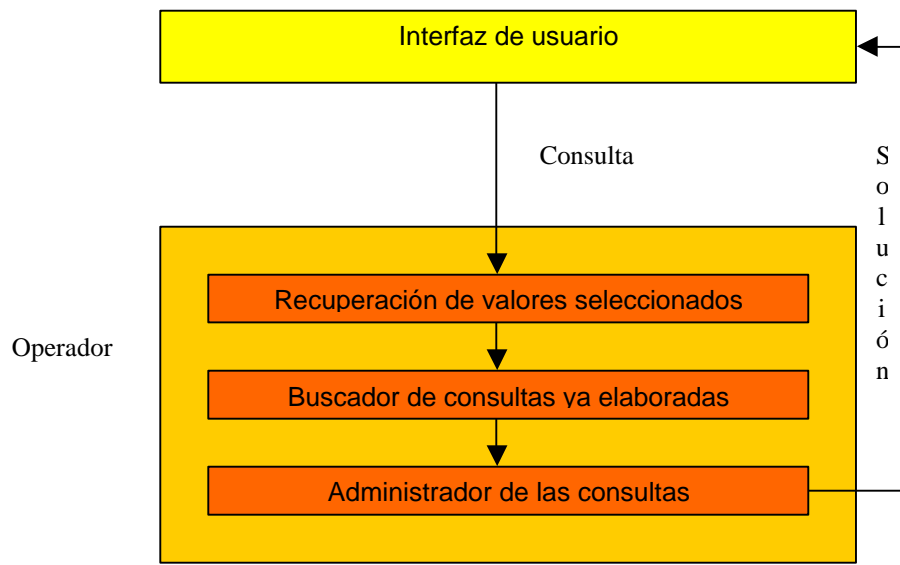


Figura 3.10 Diseño de operador *Slice 'n dice*

En la figura 3.10 se puede observar muy bien como es el diseño de este operador. De la interfaz de usuario se obtiene la consulta mediante valores seleccionados previamente por el usuario. Es decir que el usuario escoge de un cuadro combinado los valores para las dimensiones y niveles. Todos éstos parámetros, posteriormente son recuperados y agrupados para poder armar una consulta

Existen varios casos para responder una consulta:

- Se busca la consulta en la tabla de condición; si no se encuentra, quiere decir que no ha sido calculada y que se debe calcular por primera vez. Posteriormente se almacena el resultado en la tabla de consulta.

- Se busca la consulta y se encuentra la información pero en un nivel inferior. Entonces se hace una sub-consulta que recupera el valor preguntado en la consulta original.
- Se busca la consulta y ya ha sido realizada con anterioridad, entonces se recupera el resultado de la tabla de consulta.

Con éste operador se puede responder a consultas como: ¿Cuál es el máximo de metros vendidos de escocés en el año 2002 en la ciudad de Atoyatempan?, ¿Cuál es el promedio de ventas en el mes de mayo de toda la historia?

El operador *roll-up* agrega medidas que van de un nivel N_i a un nivel más general N_j de una dimensión. El diseño de éste operador es muy similar al operador *slice 'n dice* ya que una vez que el usuario ya seleccionó los niveles y dimensiones en la interfaz, la consulta los recupera y construye la consulta. Posteriormente busca en la tabla de consultas si la consulta ya fue realizada con anterioridad y verifica a su vez en la tabla de condición la dimensión y nivel que la consulta necesita. En caso de que no se encuentre la consulta o que falte algún valor, se aumenta un nivel y se realiza la consulta para posteriormente analizarla.

Por ejemplo en la dimensión de clientes, se refiere a subir del grano de ciudad al de estado o al de país. Para que se entienda mejor esto, podemos imaginar que previamente el empresario ya efectuó una consulta del promedio de ventas de los clientes de las ciudades de Atoyatempan y Atlixco. Entonces esas consultas ATEXEM ya las tiene almacenadas en las vistas materializadas. Posteriormente el empresario desea consultar el promedio de

ventas del estado de Puebla y solamente estas dos entidades lo forman. Pues bien el operador *roll-up* aumenta el nivel de agregación obteniendo el resultado de esas dos ciudades y sacando el promedio para tener como resultado el del estado. De éste modo se agiliza el proceso y no se debe repetir el trabajo que previamente ya se efectuó.

El operador *drill-down*, es la operación inversa a *roll-up*, ya que este operador utiliza consultas antes realizadas de niveles más altos para obtener resultados de una consulta de un nivel más bajo. Por ejemplo, si la primera consulta que realizó el empresario fue la del estado, el resultado se almacenó en una vista materializada previamente y si ahora necesita el promedio de ventas de los clientes de las ciudades, simplemente se baja de nivel a un nivel de ciudades que se espera hayan sido ejecutadas previamente. En caso contrario son calculados y almacenados en las vistas.

3.5 Mantenimiento

3.5.1 Mantenimiento del DW

ATEXEM diseña un refrescado por solicitud por ejemplo, el usuario decide por medio de una acción el momento preciso para refrescar los datos. Para llevar a cabo ésta actualización, se guarda la fecha en que se realizó el último refrescado. Cuando llega una solicitud de refrescado se realiza una comparación entre la fecha almacenada correspondiente al último refrescado.

3.5.2 Mantenimiento de las vistas

El procedimiento de mantenimiento de vistas es el siguiente:

Paso 1: Se identifican los datos clave de la consulta dada (medida, dimensión y nivel).

Paso 2: Si cualquiera de esos valores se encuentra en la tabla, entonces quiere decir que creció el cubo y contiene datos nuevos, hay que actualizar. Entonces para todas las tuplas contenidas en ese rango, se ingresa la tupla nueva en la tabla con el valor y se calcula para todas las referencias de la tabla consulta las cuatro funciones de agregación. Y se actualiza el resultado y la tupla ya puede ser ingresada.

Paso 3: De no ser así, simplemente se ingresa la tupla y como no a habido ningún cambio en las consultas previamente hechas no se debe actualizar nada.

Este procedimiento se encarga de comprobar que existan las tuplas en la tabla condición que contengan valores de la nueva tupla que pueda modificar los datos antes calculados, ya que de no ser así los datos almacenados en la tabla de consulta podrían no ser válidos y tener información errónea.

Por ejemplo puedo tener ya calculado el máximo de ventas de la dimensión cliente en la ciudad de Puebla, pero si entra al DW una tupla que pueda alterar ese valor, se debe de

hacer un recalcu lo ya que de lo contrario podrían perderse datos actualizados recientemente.

3.6 Discusión

Dentro del proceso de mantenimiento de ATEXEM se decidió utilizar la estrategia de refrescado por solicitud. Este proceso se llevará a cabo al iniciar ATEXEM automáticamente, además de presentar la opción de que el usuario pueda ejecutarlo cuando desee. Generalmente esta estrategia no es adecuada para ser utilizada con DW, ya que aumentaría el tiempo de espera antes de poder interactuar con el sistema. Sin embargo, dada la naturaleza de los datos y de la utilización de ATEXEM, esta estrategia es viable, ya que el sistema se utiliza diario y son pocas las modificaciones que se tienen por día.

ATEXEM es un DW capaz de soportar grandes cantidades de datos almacenados para su explotación. Se utiliza la tecnología OLAP que puede manipular la información en niveles de agregación y así el usuario poder llegar a analizar toda la información que le interesa. Es importante resaltar el objetivo del DW el cual es contar con herramientas necesarias para poder obtener indicadores clave en cuanto a la información almacenada. Sin embargo se deben tomar varios factores en cuenta, entre ellos es el tiempo de ejecución de una consulta. Para implementar los operadores OLAP se implementaron las vistas. Éstas, además de permitir la implementación de los operadores, permiten obtener resultados en menos tiempo.