Modelo dimensional que integra texto en una Bodega de Datos

MONOGRAFIA PRESENTADA PARA OPTAR AL TITULO DE INGENIERO DE SISTEMAS

Erwin Martín Alegría Velásquez
Manuel Enrique Maca Orozco

Director: Mg. Ing. Martha Eliana Mendoza

Universidad del Cauca
Facultad de Ingeniería Electrónica y Telecomunicaciones
Departamento de Sistemas – Grupo de Investigación GTI
Línea de Investigación Gestión del Conocimiento: Bodegas de Datos
Popayán, Noviembre de 2011

AGRADECIMIENTOS

La presente Tesis es un esfuerzo en el cual, directa o indirectamente, participaron varias personas leyendo, opinando, corrigiendo, teniéndonos paciencia, dándonos ánimo, acompañados en los momentos de crisis y en los momentos de felicidad.

Primeramente agradecemos a Dios por brindarnos la posibilidad de forjarnos como profesionales y llenarnos de bendiciones.

Nuestros agradecimientos a la Universidad del Cauca institución que nos forjo como personas, brindándonos la oportunidad a través del programa de Ingeniería de Sistemas para realizar nuestros estudios de pregrado.

Agradecemos a nuestra directora Mag. Martha Eliana Mendoza por haber confiado en nosotros, por la paciencia, por la dirección de este trabajo y el ánimo que nos brindó. Al Mag. Carlos Alberto Cobos Lozada por el apoyo, conocimientos y consejos compartidos durante la carrera y en la tesis de grado y a todos los Ingenieros que hicieron de una u otra forma parte de este proceso.

Gracias también a nuestros queridos compañeros, que nos apoyaron y nos permitieron entrar en su vida durante estos casi seis años de convivir dentro y fuera del salón de clase. Marsoli, Andrea, William, Melissa, Yaqueline, Lorena y Diana, gracias.

Gracias a nuestros queridos amigos por su confianza y lealtad que nos ayudaron a crecer como personas y como profesionales.

A nuestras familias por ser nuestro apoyo incondicional.

Gracias a todos.

TABLA DE CONTENIDO

ANEXOS

LISTA DE FIGURAS

LISTA DE TABLAS

LISTA DE CUADROS

1. INTRODUCCION

Las bodegas de datos (DW por sus siglas en ingles Data Warehouse)[1] hoy en día son ampliamente usadas para el análisis de grandes cantidad de datos en diversos contextos de la vida humana (mercadeo, salud, educación, investigación, entre otros). Una de las herramientas más utilizadas para consultar la información disponible en las bodegas de datos, son las herramientas de Procesamiento Analítico en Línea (OLAP por sus siglas en inglés, On Line Analytical Processing)[2], debido a su facilidad de uso y a la posibilidad que brinda de realizar análisis flexibles de los datos en tiempo real [1].

El crecimiento de internet, de la información documental en las empresas, así como de las observaciones o comentarios que se agregan en las bases de datos objeto-relacionales, aumenta día a día. Esto ha generado que la cantidad de datos no estructurados, normalmente no se use en las bodegas de datos, no porque no sea importante, sino porque los cubos de datos que se consultan a través de herramientas convencionales OLAP no permiten un apropiado manejo de dicha información. Por ejemplo, en el área de salud la información de los tratamientos médicos de un paciente, suelen ser: Nombre del paciente, medicamento recetado, fecha de la cita, diagnóstico y observaciones; este último campo (o atributo) por ser no estructurado, normalmente no se modela en una bodega de datos o en el mejor de los casos se modela como un descriptor (atributo) que no se usa en las operaciones tradicionales de OLAP (drill-down, roll-up, slice, dice, pivot). Debido a esto los analistas del negocio toman decisiones sin tener en cuenta información importante que se encuentra almacenada en los textos no estructurados, lo cual puede influir en que la decisión tomada no sea la más acertada.

Otro ejemplo ocurre en el área investigativa en la cual para revisar el estado actual de un tema en particular se debe revisar gran cantidad de documentos y algunos de ellos no aportan de la forma esperada, si se puede almacenar un artículo científico con el porcentaje de relación a ciertos temas, se permitiría que los investigadores puedan revisar los artículos que más aporten al tema que está trabajando. Estos textos no estructurados dan la posibilidad de realizar análisis más completos de la información que existe en las organizaciones.

Uno de los principales retos de las bodegas de datos y de las herramientas OLAP hoy en día, consiste en incorporar datos no estructurados y medidas textuales que permitan ser navegados en la bodega [2]. En este proyecto se propone un modelo multidimensional para una bodega de documentos que incluye medidas textuales, con una jerarquía de tópicos automática que permite que los documentos sean consultados a través de operaciones OLAP y medidas probabilísticas generadas con PLSA (Por sus siglas en inglés *Probabilistic Latent Semantic Analysis*). Adicionalmente se presenta un prototipo del sistema OLAP que utiliza artículos científicos como corpus de documentos.

A lo largo de este documento se presenta el proceso seguido para la realización del proyecto y conceptos teóricos relevantes necesarios para el desarrollo del mismo. A continuación se hace una descripción general del contenido de este documento y la organización del mismo.

[1] Por sus siglas en inglés, Data Warehouse
[2] Por sus siglas en inglés, On Line Analytical Processing

En el capítulo 1 se presenta la problemática actual que dio inicio a este proyecto, la justificación, los objetivos y los principales resultados obtenidos.

El capítulo 2 describe las bases teóricas que enmarcan el proyecto, teniendo en cuenta los conceptos de modelado en las bodegas de datos y su visualización, y estudios e investigaciones basados en el modelado de tópicos y/o términos que incorporan texto en una bodega de documentos.

En el capítulo 3 se define una arquitectura general para el procesamiento y cargue de los documentos en una bodega de documentos. Esto a partir de cuatro procesos que contienen los algoritmos Cosme, IGBHSK modificado y PLSA. También se presenta el modelo multidimensional con base en el contexto teórico de modelado, incorporando medidas textuales creadas con MDX y su respectiva función de agregación.

El capítulo 4 presenta en detalle el cargue del modelo a partir un conjunto de datos, representados en archivos de texto y XML, que contienen información de los documentos, jerarquía de tópicos y probabilidades. Además de la visualización del modelo en una herramienta OLAP que permite la navegación de los datos a través de un cubo, permitiendo a los usuarios finales (tesistas de la Universidad del Cauca) evaluar el modelo a través de una encuesta para su posterior análisis.

Los capítulos 5 y 6 describen las conclusiones que se generaron después de la culminación del proyecto y se establecen recomendaciones o posibles mejoras que se puedan incluir en un trabajo futuro para la continuidad del proyecto.

Y el capítulo 7 contiene la bibliografía y documentación empleada en la realización del proyecto.

1.1 PLANTEAMIENTO DEL PROBLEMA

En las arquitecturas actuales de bodegas de datos, los datos en su mayoría son estructurados y transaccionales, basados principalmente en hechos numéricos y orientados a métricas de negocio. Sin embargo, recientemente se ha venido incursionando en el campo de los datos no estructurados (datos de tipo texto) que se encuentran en documentos, registros médicos, correos electrónicos, hojas de cálculo, entre otros; debido a la necesidad que tienen las diferentes empresas u organizaciones en el manejo de estos dos tipos de datos, con el fin de aprovechar al máximo el conocimiento que reposa en los datos (estructurados y no estructurados) que se recolectan día a día y que en muchas situaciones se encuentran relacionados entre sí. Aunque las técnicas OLAP han mostrado ser muy útiles para el análisis de datos estructurados, se enfrentan a grandes retos en el manejo de datos no estructurados porque los cubos OLAP[3][3] que se consultan a través de herramientas convencionales de OLAP no permiten un apropiado manejo de dicha información.

Por otra parte, uno de los principales problemas que enfrenta la búsqueda de información en la web, así como los sistemas de recuperación de información (*IR* por sus siglas en inglés *Infomation Retrieval*)[4], por ejemplo en artículos científicos y tecnológicos, es la sobrecarga de información (*information overloading*). Es por esto que desde hace más de 50 años se han propuesto mecanismos sofisticados y eficientes para sumarizar (resumir) documentos de texto (datos no estructurados) [4, 5], y más recientemente se han iniciado trabajos que vinculan medidas de IR en las DW.

Entre los trabajos de investigación más importantes a nivel mundial que integran el análisis de los datos no estructurados en las DW, se encuentra Topic Cube [6] donde se propone una ampliación al cubo de datos tradicional mediante la adición de una jerarquía de tópicos y dos medidas probabilísticas de texto, permitiendo con esto, realizar sumarización de estas medidas por medio de operaciones tradicionales de OLAP, pero perdiendo el cubo la posibilidad de navegar hasta el nivel del documento (granularidad gruesa). Otro trabajo es Text Cube [7], un modelo que esboza una jerarquía de términos, comportándose como una jerarquía de organización en la que se pueden realizar las operaciones de *pull-up* y *push-down* (propias del modelo). Aquí, se aplican medidas de IR estándar que mejoran el análisis de los datos. Sin embargo, este modelo también posee problemas de granularidad en los datos no estructurados y en la unión de la jerarquía de términos.

Por lo anterior, el presente proyecto planteó la siguiente pregunta de investigación: ¿Cómo definir un modelo dimensional que integre datos no estructurados en una DW, donde el nivel de granularidad sea por documento, la navegación del modelo se apoye en una jerarquía de tópicos y la sumarización (resumen) de los documentos se pueda realizar a través de dicha jerarquía?

Teniendo en cuenta que la integración de los datos no estructurados en las DW es un área activa, abierta, de mucha importancia y con muchas áreas de aplicación, en este proyecto se propone un modelo dimensional que integra datos no estructurados en una DW, con un nivel de granularidad por documento, que permite la navegación del modelo a

[3] Los cubos OLAP son almacenes de datos. Se utilizan frecuentemente para referirse a que se están construyendo estructuras multidimensionales.
[4] Por sus siglas en inglés, Information Retrieval

través de una jerarquía de tópicos y la sumarización (resumen) de los documentos se pueda realizar a través de dicha jerarquía, facilitando su exploración por parte del usuario.

1.2 JUSTIFICACION

Actualmente existen modelos datos no estructurados en las bodegas de datos, que presentan inconvenientes o deficiencias en el manejo de este tipo de datos, como son: (1) no permiten la navegabilidad de los textos con operaciones clásicas de OLAP, es decir no cuentan con una jerarquía de tópicos integrada al modelo que permita esta navegación, ya que es definida por un experto, (2) el uso de la medidas de IR no permiten obtener un agrupamiento adecuado de los documentos de acuerdo a la jerarquía de términos, ya que estas se basan solo en técnicas de conteo, (3) no permiten que el usuario final llegue al nivel más bajo de la granularidad como lo es el documento.

El modelo dimensional propuesto pretende superar los inconvenientes o deficiencias de los modelos existentes que integran datos no estructurados en las bodegas de datos, presentando una alternativa de solución a través de un modelo hibrido a partir de los modelos *Topic Cube* y *Text Cube* permitiendo dar solución a las falencias de los mismos. Por medio de la inserción de una jerarquía de tópicos integrada directamente con el modelo, creando medidas textuales probabilísticas que faciliten el agrupamiento de los documentos así como la navegación al usuario final.

Desde el punto de vista práctico, este proyecto incorpora datos no estructurados a los ya tradicionales datos estructurados. Considere una organización que analice sus grandes cantidades de datos mediante una bodega de datos sin importar el contexto de la vida humana que representen los datos, suponga que gran parte de esa información es textual, como por ejemplo, reportes de vuelos, artículos científicos, sugerencias entre otros. Este tipo de información hace referencia en las bodegas de datos a tipos no estructurados de datos los cuales no pueden usar la potencia de las herramientas OLAP como en los tipos habituales (datos estructurados o transaccionales). En consecuencia se cuenta con un modelo que involucra el manejo de datos no estructurados, y permite la navegabilidad con las operaciones clásicas de una herramienta OLAP y el análisis de la información de este tipo de datos.

Las herramientas tecnológicas (Microsoft Visual Studio 2008 y 2010, Microsoft SQL Server 2008) usadas para realización de este proyecto fueron inicialmente seleccionadas con base en la disponibilidad de estas herramientas por parte de la Universidad del Cauca, gracias al programa MSDN Academic Alliance[5] y la experiencia de los últimos años del grupo GTI en los proyectos exitosos desarrollados con estas herramientas.

En el desarrollo de este proyecto fue necesario profundizar en temas de la carrera de Ingeniería de Sistemas como estructuras de datos (árboles, recorridos en profundidad y expansión), minería de datos y bodegas de datos, que permitieron resolver problemas técnicos como el manejo de las medidas textuales presentes en el modelo, la generación de la jerarquía multinivel entre otras. Esta experiencia permitió demostrar nuestra

[5] Acuerdo que vincula a Microsoft con entidades educativas universitarias, en la cual se permite tener acceso a software de desarrollo con propósitos académicos.

capacidad para plantear y resolver problemas en el ámbito de profesional del ingeniero y como investigadores en formación del alma mater.

1.3 CONTRIBUCIONES

Las principales contribuciones de este proyecto de grado son:

- Creación de un modelo dimensional (granularidad a nivel de documento) que integra texto a una bodega de datos. Esto debido a que los modelos propuestos anteriormente no usan una granularidad por documento lo cual no permite un buen análisis de los datos.

- Adaptación del algoritmo IGBHSK para la generación de jerarquías multinivel. Adaptar el algoritmo IGBHSK para dar soporte a una jerarquía multinivel, ya que originalmente éste algoritmo solo trabaja para un nivel.

- Creación de algoritmo Cosme. Cosme es un algoritmo que se desarrolló en este proyecto para permitir, la preparación y transformación de documentos a la bodega de datos que implementa el modelo el planteado.

- Implementación de la función de agregación. Se implementó una función de agregación que trabaja en múltiples documentos agrupados en la jerarquía a través de las diferentes operaciones OLAP.

- Creación de medida textual calculada. Se creó una medida de texto calculada para poder dar soporte a los datos no estructurados dentro de una bodega de datos.

- Prototipo de aplicación de usuario final. Se realizó un prototipo basado en una herramienta OLAP existente, que permite visualizar el modelo planteado contemplando la jerarquía de tópicos y sus probabilidades.

- Motivar el desarrollo y la investigación sobre el uso, aplicación e implementación de las bodegas de datos así como el uso de los datos estructurados y no estructurados dentro de las mismas en la comunidad académica de la Universidad del Cauca, implementando un prototipo de herramienta OLAP que cree un precedente alrededor de éste tema, y despertar así el interés por estas tecnologías en la comunidad investigativa del país, debido al positivo panorama social y económico factible de obtener con su difusión.

1.4 OBJETIVOS

1.4.1 Objetivo general

Proponer un modelo dimensional que integre texto (datos no estructurados) y datos estructurados en una Bodega de Datos que permita la sumarización de documentos a partir de una jerarquía de tópicos y medidas de IR probabilísticas con el fin de que el usuario final obtenga resúmenes de los documentos de su interés.

1.4.2 Objetivos específicos

- Definir un modelo dimensional (granularidad a nivel de documento) que integre texto a una bodega de datos que contemple:
  - Una jerarquía de tópicos que permita la exploración organizada de los documentos.
  - Medidas de IR probabilísticas que permitan la sumarización de los datos no estructurados.
  - Una función de agregación de resúmenes que trabaje en múltiples documentos agrupados en la jerarquía a través de las diferentes operaciones OLAP.
- Proponer un prototipo de aplicación de usuario final basado en una herramienta OLAP existente, que permita visualizar el modelo planteado y contemple:
  - Una jerarquía de tópicos generada automáticamente a partir de IGBHSK[6][8].
  - Medidas de IR probabilísticas generadas a partir del algoritmo PLSA [9].
- Evaluar la satisfacción del usuario con respecto a la función de agregación del prototipo propuesto, basados en los factores de facilidad[7] de uso y tiempo de consulta[8] propuestos en el modelo de Torkzadeh and Doll (1999) [10]; por medio de encuestas realizadas a tesistas de Ingeniería de Sistemas de la Universidad del Cauca.

1.5 RESULTADOS OBTENIDOS

- Modelo Dimensional con soporte a texto (datos no estructurados), junto con las implementaciones y pruebas para 200, 400 y 600 documentos.
- Algoritmo Cosme, desarrollado e implementado en este proyecto para apoyar la extracción y transformación de datos.
- Artículo: Modelo de análisis multidimensional para una bodega de documentos que incluye medidas textuales (Anexo M).
- Algoritmo IGBHSK modificado, genera una jerarquía multinivel junto con clasificación de documentos y su etiquetado.
- Prototipo OLAP para pruebas del modelo.
- Monografía del trabajo de grado. Corresponde al presente documento, donde se describe el proceso de arquitectura y desarrollo de un modelo dimensional que soporta datos no estructurados permitiendo la navegabilidad de estos datos con las operaciones clásicas OLAP, el uso de medidas de IR probabilísticas y donde la granularidad del modelo sea por documento, así como también la validación, pruebas, resultados, conclusiones y trabajo futuro.

[6] Por sus siglas en inglés, Iterative Global-Best Harmony Search with K-means
[7] Facilidad de uso: la facilidad de navegar por las dimensiones y por la medida de texto.
[8] Tiempo de consulta: tiempo de ejecución de la medida de texto.

1.6 DISEÑO METODOLOGICO

La metodología para la generación de la base conceptual está basada en el libro "Modelo Integral para un Profesional en Ingeniería" [11]. En este se plantea un modelo de Investigación Documental que se puede adaptar a las características de un proyecto en particular. Este modelo se compone de 4 fases en las cuales se desarrollan una o más actividades para cumplir con los objetivos propuestos.

- ✓ **Fase Preparatoria:** En esta fase se identificó de manera clara conceptos importantes, por medio del desarrollo de las siguientes actividades (Capítulo 2):

  - Revisión bibliográfica sobre bodegas de datos: permitió obtener el conocimiento teórico respecto a los componentes principales de una bodega, el modelamiento, los esquemas más utilizados para el diseño y el concepto OLAP para la visualización del cubo.
  - Revisión del estado del arte acerca de los modelos existentes: encontrándose dos modelos multidomensionales que dan una solución parcial al problema planteado, *Topic Cube* y *Text Cube*. Estos modelos permitieron de manera inicial proponer un algoritmo hibrido entre los modelos antes mencionados que mejora sus restricciones y características.
  - Revisión bibliográfica de los algoritmos IGBHSK y PLSA: permitió estudiar y entender a fondo los procesos y métodos utilizados por cada algoritmo, observando sus entradas de datos y las salidas correspondientes a los metadatos necesarios para el modelo multidimensional propuesto.
  - Búsqueda de herramientas OLAP para la adaptación de una aplicación de usuario final: la herramienta apta, de acuerdo a la tecnología Microsoft implementada, fue *Dundas chart* para *.net OLAP services*, siendo una herramienta de visualización e interacción con el usuario final.

- ✓ **Fase Descriptiva:** Esta fase comprendió las siguientes actividades (Capítulos 3 y 4):

  - Definición del modelo propuesto: de acuerdo a los procesos que permitieron integrar datos no estructurados en una bodega de documentos, se definió una arquitectura general que se compone de cuatro procesos, donde cada uno aporta con métodos y procesos particulares para cumplir un mismo objetivo.
  - Implementación y ejecución de algoritmo GBHSK para la creación de jerarquías para el modelo: este algoritmo fue modificado para que generara una jerarquía multinivel y pudiera ser incorporada al modelo multidimensional, específicamente definir la jerarquía de tópicos para la dimensión tópico del modelo, ya que inicialmente este algoritmo genera una jerarquía a un solo nivel.
  - Implementación y ejecución de algoritmo PLSA para creación de medidas probabilísticas para el modelo: se utilizó el algoritmo implementado por Hang Chen de la Universidad de Illinois, el cual permitió encontrar las medidas probabilísticas que se definen en el modelo multidimensional propuesto, para relacionar los documentos con los tópicos y los tópicos con las términos de los documentos.
  - Adecuación de la función de agregación necesaria para la sumarización de los datos no estructurados en el modelo: se definió la función de agregación promedio a través de un procedimiento almacenado y la utilización del lenguaje de consultas

para bodega de datos. Esta función realiza las diferentes operaciones OLAP al navegar a través de las diferentes jerarquías y dimensiones del modelo propuesto.

- Adaptación de la herramienta de aplicación de usuario final *Dundas chart para .net OLAP sevices*, en la que se llevó a cabo el prototipo del modelo.
- Selección de un conjunto de documentos científicos en formato PDF para ser procesados por los algoritmos IGBHSK y PLSA, y cargados mediante el proceso ETL.
- Evaluación de la satisfacción del usuario final con respecto a la interacción con la herramienta OLAP y la incorporación de medidas textuales en el modelo multidimensional a través de una encuesta. Adicionalmente observar el comportamiento de los usuarios en los diferentes factores encuestados y tomar el tiempo con que se ejecutan las consultas en la herramienta OLAP.
- Analizar los resultados de la evaluación mediante métodos estadísticos que permitieron deducir el comportamiento, la aceptación y la satisfacción al incorporar medidas textuales con su función de agregación en una bodega de documentos.

✓ **Fase de construcción teórica global:** En esta fase se desarrolló la actividad de elaboración del documento final y anexos, la cual se compone de un balance del conjunto de resultados de estudios, limitaciones, dificultades y logros obtenidos durante el transcurso del proyecto, identificando las ventajas y desventajas al incorporar medidas textuales con su función de agregación en una bodega de documentos.

✓ **Fase de extensión y publicación:** En esta fase se realizó la divulgación de los resultados obtenidos mediante una monografía y la presentación del proyecto.

2. CONTEXTO TEORICO

2.1 MODELO MULTIDIMENSIONAL

La necesidad de obtener información de las bases de datos u otra fuente de información para apoyar la toma de decisiones, nace junto con el desarrollo de las primeras aplicaciones, pero su importancia es reconocida mucho tiempo después. Debido a que las antiguas aplicaciones no satisfacen la necesidad de información que requiere las organizaciones para ser más competitivas y eficientes, se decide separar el procesamiento de datos en dos grandes categorías: *Operacional y Decisional*. Aparece entonces una nueva arquitectura, la cual tiene como principal componente las *Bodegas de Datos* [12].

Las bodegas de datos constituyen el centro de la arquitectura de los sistemas de información y dan soporte al procesamiento de la información con el fin de proveer una plataforma solida de datos históricos e integrados, a partir de los cuales se pueda hacer análisis. Además, estas proveen la facilidad de integración en un mundo de aplicaciones no integradas, teniendo en cuenta la organización y el almacenamiento de datos necesarios para el procesamiento y análisis sobre una perspectiva a largo plazo.

Las características que debe cumplir una bodega de datos son las siguientes [13]:

- **Orientado a sujetos o temas (temático):** solo los datos necesarios para el proceso de generación del conocimiento del negocio se integran desde el ambiente operacional, estos se organizan por temas principales de la organización para facilitar su acceso y entendimiento por parte del usuario final.

- **Integrado:** es el aspecto más importante ya que los datos dentro de la bodega están integrados, siempre y sin excepciones. La integración se refleja en la consistencia en: nombres, la codificación de estructuras, definición de atributos, entre otros.

- **Variantes en el tiempo (histórico):** todos los datos son exactos en determinado momento del tiempo. Esta característica básica de los datos en la bodega es muy diferente a los datos que se encuentran en el ambiente operacional, debido a que estos reflejan los valores del momento en el que se acceden (horizonte en el tiempo de 60-90 días), en cambio las bodegas son exactos en cualquier momento del tiempo (horizonte en el tiempo 5-10 años).

- **No volátiles (permanente):** la manipulación básica de los datos que ocurre en la bodega es mucho más simple, existen dos clases de operaciones que pueden ocurrir en una bodega, la carga inicial de los datos y el acceso a ellos. No existe actualización de los datos almacenados en la bodega como parte normal del procesamiento, ya que una vez generado los datos de la bodega no se pueden cambiar.

Debido a su orientación analítica, las bodegas presentan un procesamiento diferente, que se sustenta en un modelamiento de base de datos propio conocido como modelamiento multidimensional [14]. Este modelamiento busca ofrecer al usuario una visión respecto de la operación del negocio. De acuerdo a lo anterior el modelamiento multidimensional es una técnica para modelar bases de datos simples y extensibles al usuario final. La idea

fundamental es que el usuario visualice fácilmente la relación que existe entre los diferentes componentes del modelo. Además, en este modelo multidimensional existe una menor cantidad de tablas y relaciones, que en el modelo entidad-relación, el cual tiene ciento de tablas relacionadas entre sí y diferentes caminos para obtener una misma información, lo cual, desde la perspectiva del usuario final resulta prácticamente inusable. Dentro del entorno de las bases de datos, el modelado multidimensional es una disciplina de diseño que se sustenta en el modelo entidad-relación y las realidades de la ingeniería de texto y datos numéricos.

2.1.1 Cubo

Considere un punto en el espacio a través de sus ejes de coordenadas (por ejemplo XYZ) [15]. Un punto cualquiera en este espacio quedará determinado por la intersección de tres valores particulares de sus ejes. Al asignarle valores particulares a cada eje, por ejemplo X representa productos, el eje Y representa sucursal y el eje Z corresponde a tiempo o fecha, como ejemplo se podría obtener la siguiente combinación: producto = zapatos, región = norte, tiempo = diciembre-2010. La intersección de estos valores definirá un solo punto en el espacio. Si el punto que se busca, se define como la cantidad de zapatos vendidos, entonces se tendrá un valor específico y único para tal combinación.

Para entender un poco más el concepto, revise lo siguiente: la descripción típica de una organización es: “nosotros vendemos productos en las diferentes regiones, y nuestro desempeño se mide de acuerdo al tiempo”. Un diseñador dimensional lo verá como: “nosotros vendemos productos en las diferentes regiones, y nuestro desempeño se mide de acuerdo al tiempo”, donde cada palabra subrayada corresponde a una dimensión. Esto se puede visualizar como un cubo (ver Figura 1), donde cada punto dentro del cubo es una intersección de coordenadas definidas por los lados de éste (dimensiones).

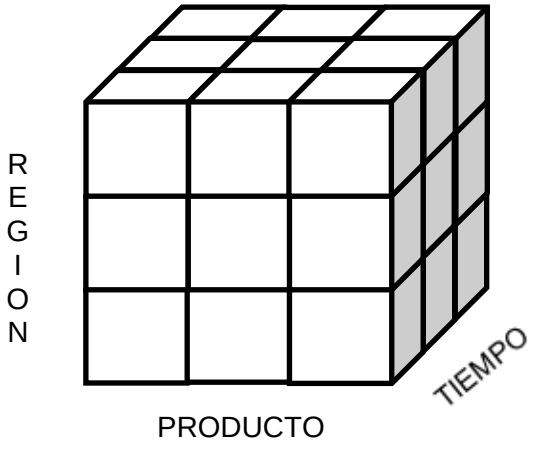


Figura 1. Cubo dimensional para la combinación de producto, sucursal y tiempo.

En el modelo multidimensional cada eje corresponde a una dimensión en particular. Entonces la dimensionalidad de la base estará dada por la cantidad de ejes (o dimensiones) que se asocien. Cuando un modelo puede ser visualizado como un cubo de tres o más dimensiones, es más fácil para el usuario organizar la información y poder

imaginar cortando o partiendo el cubo a través de cada una de sus dimensiones para obtener la información deseada.

En general, un modelo multidimensional, como su nombre lo indica, consiste en un conjunto de dimensiones que son asociados a un fenómeno de interés medible para una organización. Este fenómeno es denominado hecho. Las dimensiones se componen de niveles las cuales se estructuran jerárquicamente. De acuerdo a lo anterior el modelamiento se basa en dos componentes tablas de hechos y dimensiones [12].

- Un *hecho* es el foco de interés en el proceso de toma de decisiones, típicamente modela un conjunto de valores del mundo real. En estos se almacenan las medidas numéricas de la organización, cada medida corresponde con una intersección de valores de las dimensiones y generalmente se trata de cantidades numéricas.

- Una *medida* es una propiedad de un hecho y describe aspectos cuantitativos de interés para el análisis (por ejemplo unidades producidas, unidades vendidas, costos, ganancias, etc.).

- Una *dimensión* es una propiedad de un hecho con un dominio finito y describe una de sus coordenadas de análisis. El conjunto de dimensiones determina la granularidad o nivel de detalle de la información, ya que estas alimentan a los hechos a través de los atributos que conforman las dimensiones.

- La *granularidad* representa el nivel de detalle al que se desea almacenar la información sobre la organización que se esté analizando. Por ejemplo, los datos referentes a ventas o compras realizadas por una empresa, pueden registrarse día a día, en cambio, los datos pertinentes a pagos de sueldos o cuotas de socios, podrán almacenarse a nivel de mes. Mientras mayor sea el nivel de detalle de los datos, se tendrán mayores posibilidades analíticas ya que podrán ser resumidos.

- Una *jerarquía*, es una estructura de árbol lógica, que está compuesta por uno o varios niveles, e implica una organización de estos dentro de una dimensión. Cada nivel tiene una relación de uno-a-muchos entre objetos del nivel superior al inferior. Las jerarquías definen como los datos son agregados desde los niveles más bajos hacia los más altos.

El modelo multidimensional se puede instrumentar por un esquema relacional, este esquema almacena datos en tablas relacionales especializadas, llamadas tablas de hechos y de dimensiones. Este provee una vista multidimensional de los datos usando un modelo relacional como soporte. Los hechos son almacenados en la tabla de hechos así como las dimensiones en una tabla de dimensiones. Estos se encuentran ligados entre sí, es decir, la tabla de hecho se encuentra ligada a sus dimensiones.

2.1.2 Esquema estrella

En el esquema estrella [16], una sola tabla de hechos está relacionada a cada tabla de dimensión. Las tablas de dimensiones son enlazadas a la tabla de hechos mediante

referencias de una llave foránea. La llave primaria en la tabla de hechos se compone de una relación de las llaves primarias de las tablas de dimensiones. En la
Figura **2** se presenta una tabla de hechos y tiene asociado tablas de dimensiones, cada una de estas tablas tiene un identificador único, el cual corresponde a la clave de identificación en la tabla de hecho.

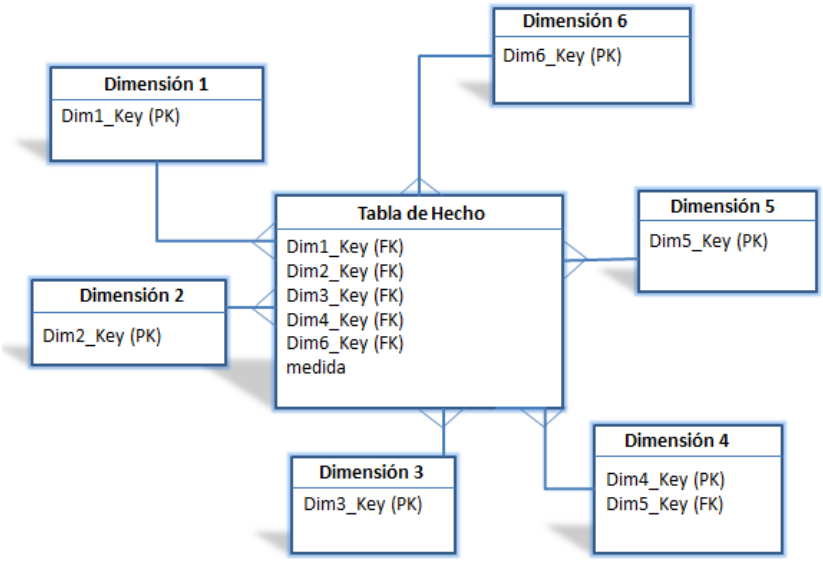


Figura 2. Ejemplo esquema estrella

2.1.3 Esquema copo de nieve

El esquema estrella puede ser redefinido en el esquema copo de nieve [16] con un soporte para jerarquía de atributos, permitiendo que las tablas de dimensiones tengan tablas de sub-dimensiones. En este esquema las tablas son normalizadas para simplificar las operaciones de selección de datos, con lo que logra presentar la información sin redundancia, evitando anomalías en los datos. Este esquema representa mejor la semántica las dimensiones de ambientes de los negocios, ya que tiene un acceso más directo a los datos, lo cual se traduce en una eficiente recuperación de la información que manipulan las tablas. La desventaja de normalizar las tablas es que hace más complejo el proceso de cargue de la información (poblar y administrar), adicionalmente muchas uniones en las consultas puede sacrificar el desempeño del sistema, el tiempo de consulta y la visualización al usuario. Este último debido a que hay un mayor número de dimensiones aumentando la complejidad en las consultas [17]. La Figura 3 muestra un ejemplo similar al anterior con el mismo número de dimensiones, la única diferencia es la tabla de Dimensión 4, la cual maneja a su vez sub-dimensión permite un acceso más rápido a la Dimensión 5 que maneja.

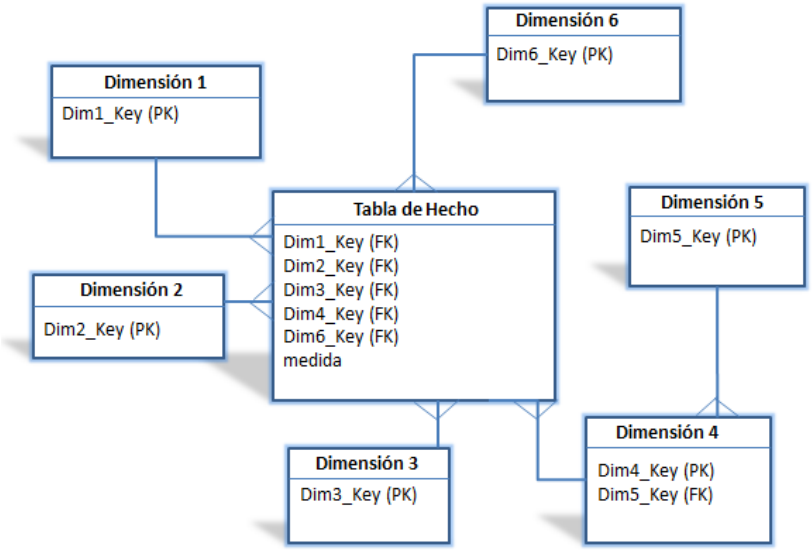


Figura 3. Ejemplo esquema copo de nieve

2.1.4 OLAP

Una bodega de datos no es útil para la organización sino cuenta con una herramienta apropiada para su consulta. Las herramientas OLAP (Online Analytical Processing) han sido por años, las herramientas usadas para consultar las Bodegas de datos y han mostrado ser apropiadas para que los usuarios exploren en forma adecuada los datos que reposan en ellas. Las herramientas OLAP se basan en una vista multidimensional de los datos, y esto ha sido la clave de su éxito, en principio porque define un modelo "sencillo" de diseño que es comprensible y fácil de usar por parte de los usuarios que toman las decisiones en las organizaciones [18, 19]. Las herramientas OLAP presentan al usuario una visión multidimensional de los datos para cada actividad que es objeto de análisis, es decir cuando un usuario formula consultas a la herramienta OLAP, selecciona atributos del modelo multidimensional sin necesidad de conocer la estructura interna o el esquema físico de la bodega de datos. Una consulta a una bodega de datos consiste generalmente en la obtención de medidas sobre los hechos parametrizados por atributos de las dimensiones y restricciones por condiciones impuestas sobre las dimensiones.

Entre las operaciones básicas de OLAP [20] se encuentran: (i) *slice*, se refiere a la selección de las dimensiones para generar una vista del cubo (similar al operador selección del álgebra relacional); (ii) *dice*, hace referencia a la selección de valores sobre una dimensión (similar al operador proyección del álgebra relacional); (iii); *drill-down*, se refiere a decrementar el nivel de agregación a través de una o más jerarquías de dimensión y (iv) *roll-up*, la operación contraria a *drill-down*, permite incrementar el nivel de agregación; (v) *pivot*, hace referencia a rotar las dimensiones para proveer una presentación alternativa de los datos. Las operaciones *slice* y *dice* permite reducir la dimensionalidad de un cubo.

2.2 BODEGAS DE DOCUMENTOS

Por muchos años, las investigaciones sobre OLAP y las bases de datos multidimensionales han generado metodologías, herramientas y sistemas de administración para el análisis de datos estructurados (datos numéricos). Con el crecimiento de los documentos digitales, se ve la necesidad de incorporar texto enriquecido dentro de las bases de datos multidimensionales como un marco de trabajo adaptado para su análisis. Las investigaciones realizadas sobre el análisis de textos con herramientas OLAP se pueden agrupar en dos: el texto como dato categórico y el texto como un componente de OLAP.

2.2.1 Texto como dato categórico

En el enfoque del texto como un dato categórico se aplican métodos de clasificación de los documentos en etiquetas de clase dentro de una categoría. Este modelo permite operaciones de drill-down o roll-up sobre la dimensión de categoría pero usa un cubo multidimensional tradicional (sólo con datos estructurados), además para el proceso de clasificación en estas propuestas se necesitan datos de entrenamiento, es decir, documentos previamente asignados a las categorías, y estos datos no están disponibles en todos los escenarios.

En este enfoque existen varios trabajos representativos, el de Cody et al. de IBM en el 2002 [21], que proponen algunas ideas generales de cómo integrar algoritmos de minería de textos con OLAP, sin embargo, en este trabajo los cubos siguen siendo los tradicionales y con un bajo nivel de integración (a nivel externo) de las dos tecnologías. Además, no se trata el tema de la materialización eficiente de los cubos que manejan la dimensión de texto.

También se encuentra el trabajo de McCabe [22] en el 2000 enfocado en recuperación de la información, que presenta la ocurrencia del término en el documento como un hecho del modelo multidimensional (ver Figura 4), con dimensiones estándar para el término y el documento, dimensiones con jerarquía de organización (relación padre hijo) la fecha y la localización; y una dimensión categoría que contiene una temática o una jerarquía. La medida es un peso que es el valor del término dentro del documento (frecuencia del término). El objetivo principal de esta investigación consiste en realizar búsquedas por los términos incluidos en el documento, combinando con fecha y localización.

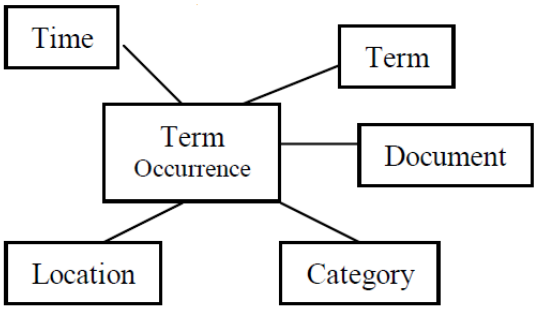


Figura 4. Esquema estrella, modelo de datos multidimensional (McCabe)

Por este mismo camino Tseng y Chou [23] en el 2006 presentan un marco de trabajo general para una bodega de documentos (fuentes de datos, componentes de interface y del servidor), y definen un modelo (ver Figura 5) general para este tipo de bodegas, clasificando las dimensiones en tres categorías: dimensión normal con el conjunto de palabras clave que permita a los usuarios localizar los documentos deseados directamente, dimensión de conjunto de datos que dan información del documento, como: título, autor, publicador, fecha, etc. Por último la dimensión categoría, que puede ser una jerarquía basada en WordNet o definida por el usuario. El hecho se compone de una clave compuesta por las llaves foráneas de las dimensiones nombradas anteriormente, atributos usados para derivar las medidas del documento (frecuencia del término), una columna Documento_ID que representa el identificador del documento que es una clave foránea de una dimensión que contiene todos los identificadores de los documentos y la ruta del archivo. En este modelo la granularidad de la tabla de hechos es por palabra clave, lo que hace que la navegación del usuario por el documento sea muy compleja, además solo contempla medidas de conteo, ninguna medida probabilística que permita dar una mejor aproximación y relevancia de los temas que trata el documento.

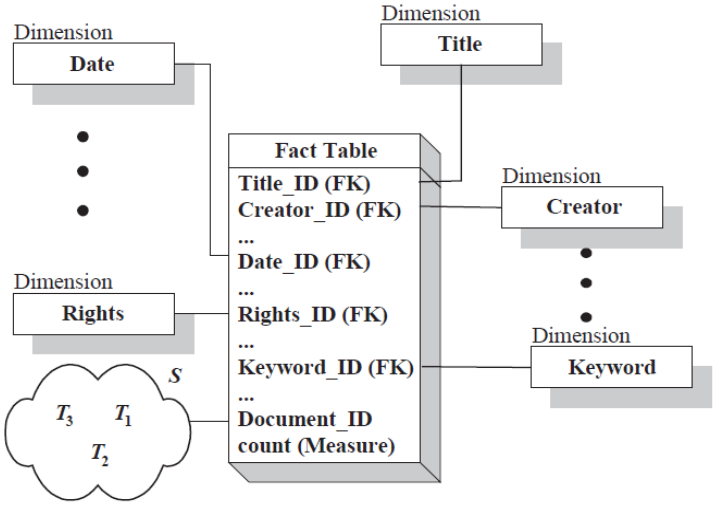


Figura 5. Esquema estrella de una bodega de documentos (Tseng y Chou)

Por su parte Franck Ravat et al. [24] en el 2007, proponen un modelo multidimensional donde integran una función de agregación para el análisis de las palabras clave de una publicación con el fin de proveer una vista del contenido de la publicación. Para permitir el análisis OLAP de documentos, proveen una función de agregación diseñada para agregar conjuntos de palabras clave, por ejemplo dado un conjunto de palabras clave como entrada, la función genera un nuevo conjunto de palabras clave agregadas, este proceso de agregación está basada en un modelo conceptual que provee, 1) conceptos o temas adaptados para soportar la medida textual no-numérica así como una representación jerárquica de los conceptos analizados con el uso de ontologías y 2) un nuevo concepto de unidad de procesamiento de agregación textual OLAP con el uso de ontologías de dominio. El objetivo es proveer una función de agregación adaptada para las medidas textuales. Sin embargo, únicamente las funciones de agregaciones genéricas operan

sobre cada una de las medidas, lo cual no es suficiente para el análisis multidimensional. Con el fin de manipular medidas textuales elaboradas, definieron una función de agregación inspirada en la función promedio (AVERAGE). Esta función agrega un conjunto de palabras clave dentro de un pequeño conjunto de palabras claves más generales.

De acuerdo a lo anterior la función de agregación extrae de una medida textual (compuesta por n palabras) un conjunto de *k* términos más representativos. Así como en la función de agregación MAX, extrae el valor más alto de un conjunto de números, la función de agregación propuesta extrae el termino *k* más representativo de un texto compuesto de un conjunto de n palabras. Concretamente la función ordena todas las palabras que representan el documento de acuerdo a un peso asignado correspondiente a su representatividad respecto al documento y retorna los primeros k términos con sus pesos, estos pesos se obtienen al aplicar la función *tf-idf* (tf: frecuencia del término, idf: frecuencia inversa del documento) a un conjunto de documentos cuyas palabras han sido extraídas.

El análisis OLAP se hace a través de las tablas multidimensionales, donde los valores se ubican en celdas c_{ij} que son la intersección de la i-esima fila y la j-esima columna. Cada celda contiene el valor agregado de las medidas analizadas. Para cada celda de la tabla multidimensional, la función de agregación es aplicada. Cada celda representa un cierto número de documentos o fragmentos de los documentos. Para cada celda c_{ij} corresponde: Un conjunto de documentos D_{ij} compuesto de d_{ij} documentos y Un número total de términos n_{ij} que están en cada uno de los documentos d_{ij}. Dentro de cada celda, los pesos son asignados para cada uno de los n_{ij} términos de acuerdo al "ranking" de los términos, donde la función *tfidf* (ver ecuación 8) corresponde al producto de la representatividad del término en el documento con la inversa de su representatividad en todos los documentos disponibles de la colección. Adaptando esta función al contexto, es decir, la *idf* es calculada únicamente para los documentos de la celda, se tiene que para cada celda c_{ij} y cada término t corresponde:

- Un numero de ocurrencias $n_{ij}(t)$ del término *t* en el documento de c_{ij}, es decir, D_{ij};
- Un número de documentos $d_{ij}(t)$ que contiene el termino *t* entre los documentos de c_{ij} $(d_{ij}(t) \leq d_{ij})$

Así se obtiene, para cada término de la celda c_{ij}una función *tfidf* adaptada:

$$tf_{ij}(t) = \frac{n_{ij}(t)}{n_{ij}} \; and \; idf_{ij}(t) = log \frac{d_{ij}+1}{d_{ij}(t)}, \qquad (7)$$

Así el peso del término t es: $w_{ij}(t) = tf_{ij}(t) \times idf_{ij}(t), \qquad (8)$

Este modelo aunque es muy interesante, requiere de un manejo de ontologías, que lo hace más complejo, ya que este corresponde a una jerarquía de dominio de conceptos, donde cada nodo de la jerarquía representa un concepto y cada arista entre nodos modela una relación más compleja que es una relación "es-un". Nuestro modelo difiere de estos trabajos en que introduce medidas del contenido del documento diferentes a las tradicionales tf e idf, la granularidad de la tabla de hechos (TH) es a nivel de documento y

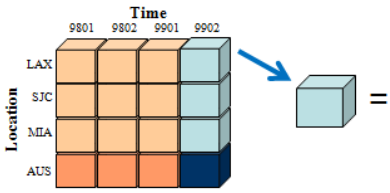
no de los términos (palabras clave del documento) como se plantea en estos trabajos, con lo cual se permite mayor poder de consulta sobre los datos; y además la función de agregación de la medida textual solo utiliza conceptos de MDX y de un lenguaje de programación clásico.

2.2.2 Texto como un componente OLAP

En los últimos años ha sido de mucho interés el enfoque del *texto como un componente de OLAP*, que utiliza la tecnología OLAP para explorar datos de texto que están almacenados en una base de texto multidimensional (MTD por sus siglas en inglés, Multidimensional Text Database). Uno de los principales aportes se enmarca en los Sistemas de Gestión de Contenidos (CMS) con la propuesta de Alkis et al. [25] en el 2008, denominada Multidimensional Content eXploration (MCX). En MCX se usa un repositorio de documentos de un CMS y cuando el usuario realiza una consulta, se mezclan técnicas de RI y OLAP para dar reportes del tipo "los k resultados más relevantes" junto con resúmenes de los textos a través de una dimensión plana de términos. Este trabajo se enfoca en el proceso eficiente de los cubos, ya que la consulta puede involucrar cientos, miles o millones de documentos. Para extraer los términos más representativos de la colección, usa los términos o frases más frecuentes a la consulta en tiempo real y basado en un muestreo aleatorio de los documentos, que lo hace viable (escalable) en consultas con muchos documentos. Para el análisis de los documentos, el cubo multidimensional se construye en tiempo de ejecución y se modelan dos tipos de dimensiones, unas estáticas, previamente conocidas en el área de aplicación y unas dinámicas, que se basan en los términos o frases almacenados en los documentos. Los principales inconvenientes de MCX están en que los términos o frases frecuentes no tienen el significado suficiente para los usuarios y que no permite una exploración jerárquica de los tópicos o temas de la colección, lo que dificulta el análisis por parte de los expertos.

Después de MCX, en el 2008 se propuso un modelo denominado "Text Cube" [26] (Ver Figura 6) que mezcla técnicas tradicionales de RI, como el cálculo de TF-IDF [27, 28], la remoción de palabras vacías, la lematización[9] de las palabras, el uso de un vocabulario específico en el área de aplicación, WordNet como ontología de nivel general para convertir términos en conceptos y un índice invertido (IINV) de documentos. Con todo lo anterior registra TF y el IINV en una dimensión del cubo, permitiéndole con ello sumarizar (tema de trascendental importancia en OLAP) y navegar, utilizando las principales operaciones OLAP sobre dicha dimensión. En esta propuesta un Text Cube se compone de medidas tradicionales de OLAP y de medidas relacionadas con RI, además de jerarquías de dimensión (tradicionales en OLAP y sobre las cuales se puede hacer drill-down, roll-up, slice y dice) y de unas nuevas jerarquías de términos (dadas por expertos del dominio de aplicación y en la que se especifican los niveles semánticos y las relaciones entre los términos del texto).

[9]Es el proceso mediante el cual se relacionan morfológicamente las palabras que comparten la misma raíz, de forma que se pueda agrupar las variantes morfológicas de cada término (ej. blog, blogs, blogging, bloggers).



Term	TFIDF
flight	35.6
weather	20.9
LA	11.5
land	6.5
taxi	3.5
…	…

Dimensions				Text Data
M (Model)	P (Price)	T (Y/M/D) (Time)	S (Score)	$\mathcal{DOC}$ (Documents)
m1	p1	2007/07/01	s1	$d_1 = \{w1, w1, w2, w6, w8\}$
m1	p1	2007/07/01	s2	$d_2 = \{w1, w3, w6, w6, w7\}$
m1	p2	2007/08/01	s2	$d_3 = \{w2, w3, w6, w6\}$
m2	p2	2007/08/01	s2	$d_4 = \{w4, w5, w6, w7\}$
m2	p3	2008/06/01	s1	$d_6 = \{w4, w4, w5, w8\}$

Relevant Dimensions		Aggregated Text Data	IR Measure
M	S	D	F(D)
m1	s1	$\{d_1\}$	…
m1	s2	$\{d_2, d_3\}$	…
m2	s2	$\{d_4\}$	…
m2	s1	$\{d_5, d_6\}$	…

Figura 6. Ejemplo de un "Text cube" (Tomado de [26])

Algunos de los grandes aportes que realizan con este modelo se deben en gran parte a la inserción de algunos conceptos nuevos como son:

- **Jerarquía de términos:** es una jerarquía para especificar los niveles semánticos de las relaciones entre los términos del texto (ver Figura 7).

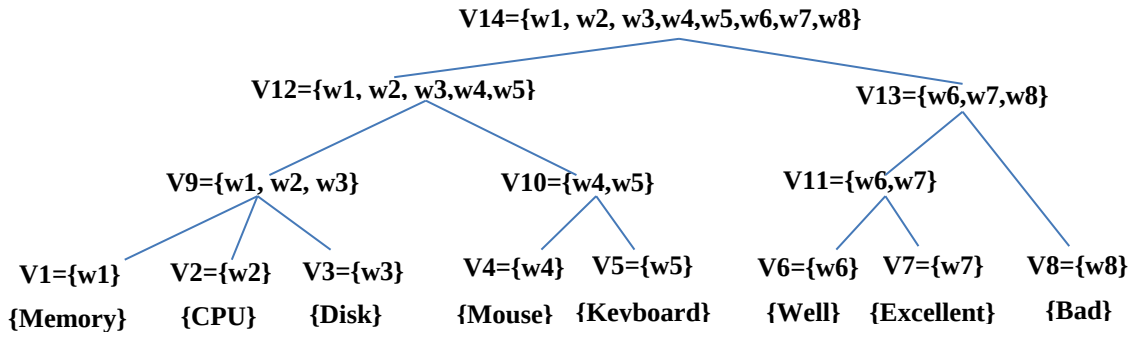


Figura 7. Jerarquía de términos "Text cube" (Tomado de [26])

- **Term frecuency (TF):** la frecuencia de los términos es una medida que se usa en IR con la cual cada término se hace corresponder con la frecuencia (absoluta o normalizada) con que aparece en el documento.
- **Inverted index (IV o IDF):** el índice invertido es una medida que se usa en IR con la cual se obtiene el número de veces que aparece el termino de mayor frecuencia en todos los documentos.
- **Pull-up:** operación OLAP para navegar hacia arriba por la jerarquía de términos.
- **Push-Down:** operación OLAP para navegar hacia abajo por la jerarquía de términos.

En este modelo la Jerarquía de términos, se comporta como una jerarquía de organización (la profundidad de la jerarquía puede variar de una rama a la otra) en la que

se pueden realizar las operaciones de pull-up y push-down (propias del modelo que no son soportadas por herramientas OLAP convencionales). Esta jerarquía se usa para calcular los sumarios (resúmenes) de las celdas[10], pero al no existir una dimensión relacionada con esta jerarquía en la tabla de hechos (TH), no se permite explorar los datos de texto con operaciones OLAP convencionales. Esta propuesta presenta ciertas desventajas: (i) El modelo no permite la navegabilidad de los textos (al no contar con una jerarquía de tópicos integrada directamente al modelo) con operaciones clásicas de OLAP, (ii) las medidas estándar de IR no permiten obtener una mejor clasificación de los documentos en la jerarquía de términos, ya que estas se basan en técnicas de conteo. En la presente propuesta se pretende solucionar estas limitaciones, integrando una jerarquía de tópicos directamente al modelo con sus operaciones clásicas OLAP, también se incluirán medidas de IR probabilísticas las cuales permiten obtener un mejor rango de posibles opciones de clasificación de los documentos en la jerarquía de términos.

Luego Zhang et al. en el 2009 [29, 30], proponen un nuevo modelo de datos llamado "Topic Cube". Este modelo permite ampliar las bodegas de datos tradicionales, adicionando una jerarquía de tópicos, medidas probabilísticas del texto y las operaciones clásicas de las herramientas OLAP sobre las dimensiones de texto construidas. La nueva dimensión de textos, permite analizar los cubos desde jerarquías de tópicos o temas textuales que son organizados con una implementación mejorada del algoritmo EM[11] [31] de agrupamiento (ver Figura 8).

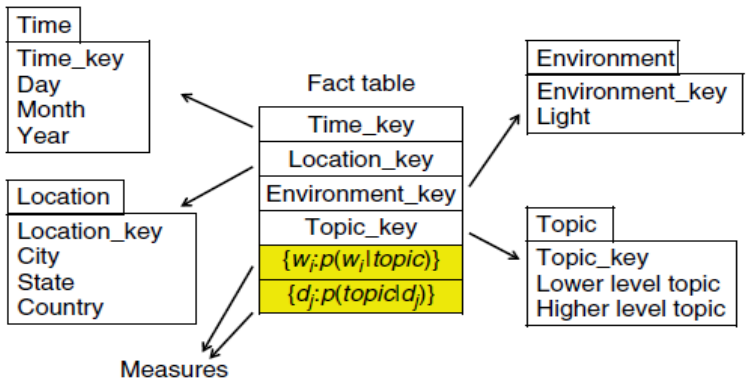


Figura 8. Esquema estrella del "Topic cube" (Tomado de [30])

En la Figura 9 se muestra un ejemplo de la materialización del cubo con la dimensión Ubicación (Location), Tiempo o Fecha (Time) y la dimensión Tópico (Topic), que cuenta con una jerarquía en donde las operaciones estándar de drill-down y roll-up se pueden realizar en cualquier dimensión del modelo.

[10] La intersección de las dimensiones en el cubo se denomina celda. Una celda puede tener más de un valor.
[11] Por sus siglas en inglés, Expectation - Maximization

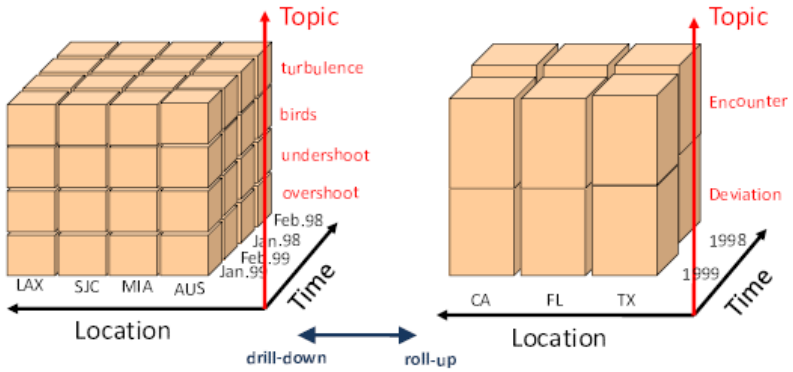
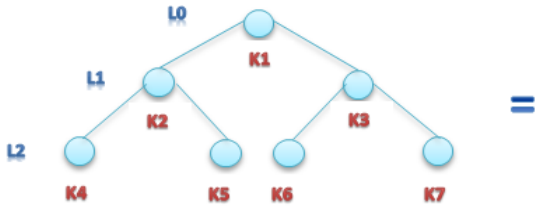


Figura 9. Un ejemplo de un "Topic cube" (Tomado de [30])

El proceso de construcción de un *topic cube* se realiza en dos grandes pasos que se detallan a continuación:

1) Modelado de un Topic Cube: Un experto debe definir un árbol o jerarquía balanceada de tópicos (*K1, K2*, etc.) que permite clasificar los temas de acuerdo al tipo de negocio (ver Figura 5) y donde los niveles del árbol son *L0, L1*, etc. A partir de esta jerarquía se define una dimensión tópico que permite a los usuarios ver los datos en las diferentes granularidades. En la Figura 10 se observa cómo queda almacenado en la dimensión tópico (*Dim_Topic*) el nodo *K3* del árbol de la jerarquía de tópicos.



=

Dim_Topic		
Topic_Key	Lower level topic	Higher level topic
1	K7	K3
2	K6	K3
3	K5	K2
…	…	…

Figura 10. Árbol de jerarquía de tópicos y Dimensión Tópico. Se definen Li niveles y en cada nivel hay Ki tópicos

En la tabla de hechos se adicionan dos medidas probabilísticas (ver Figura 8):

- **p (wi|topic):** La probabilidad de las distribuciones de palabras w_i dado un *tópico*. Conociendo el tópico, se puede conocer la probabilidad de que w_i esté en el tópico.
- **p (topic|dj):** La probabilidad de un *tópico* dado un documento d_j. Conociendo el documento d_j, se puede conocer la probabilidad de que el *tópico* esté en el documento d_j.

2) Para la materialización de un *Topic Cube* se definen dos estrategias: (i) Estrategia *Bottom-Up* y (ii) Estrategia *Top-Down*. La idea principal es aprovechar los tópicos descubiertos por las sub-celdas para obtener un buen punto de partida para descubrir los tópicos en una celda superior con PLSA[12] [9].

[12] Por sus siglas en inglés, Probabilistic Latent Semantic Analisys

(i) Estrategia Bottom-up: Construye un topic cube a partir de los primeros tópicos calculados en las sub celdas pequeñas y agregándolas a las super celdas grandes. Tal algoritmo de materialización heurística puede ser implementado de tres maneras complementarias para agregar los tópicos: Agregación a lo largo de las dimensiones estándar, Agregación a lo largo de las dimensiones del tópico y la combinación de las dos estrategias.

(ii) Estrategia Top-Down: Inicialmente se buscan todos los tópicos en el árbol jerárquico de las celdas superiores en el cubo de datos tradicional. Luego con el fin de calcular los tópicos en las sub celdas de una celda superior, se deben usar las distribuciones de palabras calculadas de tópicos en la celda superior como punto de partida y buscar tópicos en las sub celdas individualmente. Después de que las sub celdas son materializados, se puede calcular la sub sub celda de la celda superior y las distribuciones de palabras de los tópicos en sus respectivas súper celdas que se utilizarán como puntos de partida. Este proceso continúa iterativamente hasta que todo el Topic Cube se materialice.

Este modelo requiere contar previamente con la base de datos de textos y la jerarquía de temas o tópicos. Luego, se mapean todos los documentos a la jerarquía de tópicos y posteriormente se computa una medida del documento en cada celda. Aquí se presenta ciertas restricciones: (i) la jerarquía de tópicos debe ser definida por un experto para que la estrategia de materialización del cubo sea eficiente, por otra parte si no se define una jerarquía de tópicos el algoritmo de materialización que utiliza este modelo puede tardarse mucho en converger, (ii) no le permite al usuario final llegar al nivel más bajo de granularidad como lo es el documento. Teniendo en cuenta estas restricciones, la propuesta pretende generar la jerarquía de tópicos a partir de un algoritmo en particular, el cual obtendrá la información requerida para la materialización sin tener en cuenta a un experto en el área, además el modelo permitirá que el usuario pueda llegar al nivel más bajo de granularidad como lo es el documento.

Por su parte iNextCube [32] es un demo en la web[13] que permite la búsqueda y el análisis de bases de datos de texto multidimensional, proponiendo un modelo de cubo OLAP que permite la sumarización y la navegabilidad de datos estructurados junto con los datos no estructurados basados en estudios previos (Topic Cube y Text Cube). Debido a los cuellos de botella encontrados en Topic Cube y Text Cube para el descubrimiento de conocimiento y el análisis de redes de información se propuso un método que ayuda a la construcción de dichas jerarquías automáticamente llamado RankClus [33] y su extensión NetClus [34].

Para promover el diseño sistemático, el desarrollo de métodos escalables y eficientes para la búsqueda de conocimiento, iNextCube integra dos temas de investigación, texto OLAP y el análisis de redes de información en un sistema basado en internet llevando a cabo las siguientes tareas: (i) clasificación automatizada, agrupación y generación de jerarquías por NetClus, (ii) la construcción de Topic Cube y Text Cube en el sistema iNextCube y (iii) minería de datos en iNextCube.

Arquitectura de iNextCube: iNextCube tiene una arquitectura de cuatro capas, como se muestra en la Figura 11. La capa inferior intermedia es la NetClus, módulo que analiza las

[13]http://inextcube.cs.uiuc.edu

redes de información y genera las agrupaciones, las clasificaciones y las jerarquías. La capa superior intermedia está constituida de las infraestructuras de Text Cube y Topic Cube, que ofrecen en línea las medidas de recuperación de información y los parámetros del análisis semántico latente probabilística (PLSA). La capa superior interactúa con los usuarios y responde a sus peticiones, con una interfaz y visualización de usuario amigable y la capa inferior corresponde a las bases de datos de texto multidimensional.

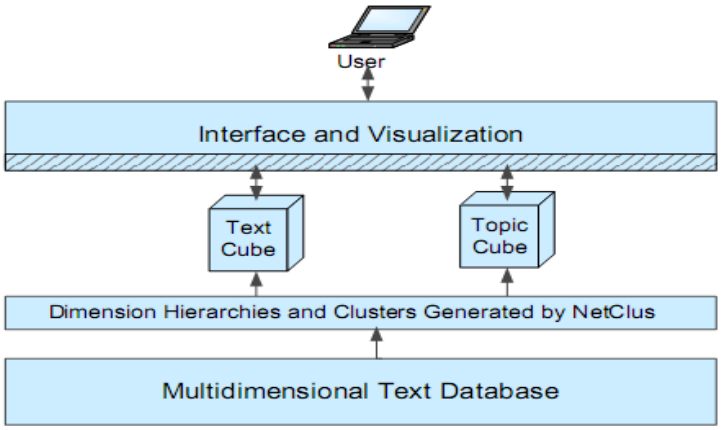


Figura 11. Arquitectura de iNextCube (Tomado de [13])

Los principales módulos funcionales son:

- La formación de la jerarquía de dimensiones basada en el análisis de redes de información a partir de RankClus y NetClus.
- Búsqueda de la información a través de Topic Cube y Text Cube, operaciones OLAP y minería de texto multidimensional.

En este modelo de composición no necesariamente se tiene que contar con una jerarquía ya definida ya que por medio de RankClus se puede generar, el cual les permitirá a los modelos Text Cube y Topic Cube el análisis sobre las bases de texto multidimensional de manera independiente ya que RankClus solo usa estos modelos para las búsquedas y mostrar los resultados de cada caso. Como restricciones se tienen: (i) se heredan las restricciones relacionadas a los modelos de análisis Text Cube y Topic Cube, debido a que no se propone un nuevo modelo de análisis que combine los ya existentes, (ii) las interfaces de visualización al usuario final no permiten que los resultados que se obtienen a partir de las consultas, sean de fácil comprensión.

Finalmente es de resaltar que el presente trabajo de grado toma como referente la jerarquía de tópicos utilizado por Topic Cube y la granularidad por documento propuesto en Text Cube, con lo cual se busca solucionar las diferentes restricciones presentadas en los modelos antes mencionados, obteniendo así, un modelo dimensional hibrido capaz de extraer información que represente conocimiento de interés para el usuario final, además de permitir navegar por una jerarquía de tópicos y resumir hasta el nivel del documento incorporando texto como medida textual en el modelo multidimensional.

2.3 PLSA

El Análisis Semántico Latente Probabilístico [35] (PLSA, por sus palabras en inglés, *Probabilistic Latent Semantic Analysis*) se deriva de una visión estadística de LSA. El análisis semántico latente [36] (LSA por sus siglas en inglés *Latent Semantic Analysis*) toma la representación del espacio vectorial (dimensional) de documentos basados en la frecuencia de términos (tf) como punto de partida para realizar una reducción de dimensionalidad a un espacio vectorial llamado espacio semántico latente, esta reducción se basa en una descomposición de valores singulares (*SVD* por sus siglas en ingles *Singular Value Decomposition*) de la matriz de términos por documentos correspondiente. El objetivo es la de representar relaciones semánticas entre palabras y/o documentos en función de su proximidad en el espacio semántico.

Aunque se encontró que el desempeño práctico de LSA es bueno en muchas tareas de recuperación de la información, se ha demostrado que posee defectos [35, 37] además del problema de selección de dimensionalidad. Los problemas encontrados son de dos tipos: (i) se deben detectar los sinónimos para que se verifique realmente la similitud de los documentos. (ii) se tiene que hacer frente a la polisemia[14] para evitar sobreestimar la similitud real entre los documentos, contando términos en común que se utilizan en diferentes significados. Ambos problemas pueden conducir a un puntaje de similitud de léxico inapropiado que no refleja la "verdadera" similitud oculta en la semántica de las palabras. Para resolver estos problemas, Hoffman [35] propuso PLSA, una extensión probabilística de LSA y basado en un modelo estadístico que se conoce como un modelo de aspecto.

Dado un conjunto de documentos $D=\{d_1, d_2, d_3,\ldots, d_n\}$ con palabras en un vocabulario $W=\{w_1,w_2,w_3,\ldots, w_m\}$, ignorando el orden o la secuencia con que ocurre una palabra en un documento, se puede resumir los datos observados en una matriz A de co-ocurrencia documento-término, donde el elemento A_{ij} indica el número de veces que el termino w_j ocurre en un documento d_i. Esta representación de colecciones de documento es llamado modelo espacio vectorial y es altamente usado por muchos algoritmos en recuperación de textos basado en palabras clave.

PLSA introduce la variable semántica latente $Z=\{z_1, z_2,\ldots, z_k\}$ con cada observación. La Figura 12 muestra la base del modelo probabilístico. El modelo conocido como *modelo de aspecto*, tiene dos formas: la forma *asimétrica* y la forma *simétrica*. Para la presente investigación se usó la forma asimétrica, ya que como lo sugiere Hoffman si la cardinalidad de z es menor que le número de documentos el modelo puede ser parametrizado de forma asimétrica.

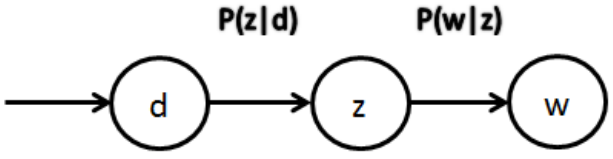


Figura 12. PLSA de forma asimétrica

[14] Polisemia: cuando una palabra o signo lingüístico tienen varios significados

Un modelo generativo para términos/documentos co-ocurrentes se puede definir por el siguiente esquema:

1. seleccionar un documento *d* con probabilidad *P (d),*
2. escoger una clase latente *z* con probabilidad *P (z|d),*
3. generar una palabra *w* con probabilidad *P (w|z).*

Como resultado se obtiene un par observado*(d, w)*, mientras que la variable de clase latente z se descarta. Dado el modelo de aspecto, la probabilidad conjunta sobre la matriz *A=DxW* se define como:

$$P(d,w) = \sum_{z\epsilon Z} P(z)P(w|z)P(z|d), \quad (1)$$

De esta forma, la relación entre dos documentos puede ser expresado por *P(z|d)* y *P(z),* en lugar del original *P(d, w)*. Las salidas del algoritmo de PLSA es la mejor estimación *P(d|z), P(w|z)* y *P(z).* Siguiendo el principio de *verosimilitud (likelihood),* se determina la mejor estimación de maximización de la función *log-likelihood* (ver Ecuación 2).

$$L = \sum_{d\epsilon D} \sum_{w\epsilon W} n(d,w) \log P(d,w), (2)$$

Este algoritmo puede ser implementado usando el algoritmo *Esperanza-Maximización*(*EM* por sus siglas en ingles *Expectation-Maximization* [38]). EM alterna dos pasos: un paso de esperanza (*paso E*), donde se calculan las probabilidades posteriores *P(z|d,w)* para las variables latentes de *z*, con base en las estimaciones actuales de los parámetros, y un paso de maximización (*paso M*), donde los parámetros *P(z|d),P(w|z)* y *P(z)* se actualizan para las probabilidades posteriores calculadas en el paso anterior E, el algoritmo se repite hasta converger.

En otras palabras, el paso E calcula las probabilidades de la Ecuación 3.

$$P(z|d,w) = \frac{P(z)[P(z|d)P(w|z)]}{\sum_{z'} P(z')[P(z'|d)P(w|z')]'} \quad (3)$$

Donde *P(z|d,w)* es la probabilidad posterior de la variable latente z, dada la ocurrencia de una palabra w en el documento d y los parámetros *P(z|d), P(w|z)* y *P(z)* son el resultado de la última iteración EM.

Y el paso M actualiza los parámetros para las ecuaciones 4, 5 y 6.

$$P(w|z) = \frac{\sum_d n(d,w)P(z|d,w)}{\sum_{d,w'} n(d,w')P(z|d,w')} \quad (4)$$

$$P(z|d) = \frac{\sum_w n(d,w)P(z|d,w)}{\sum_{d',w} n(d'|w)P(z|d',w)} \quad (5)$$

$$P(z) = \frac{\sum_{d,w} n(d,w)P(z|d,w)}{R}, (6)$$

$$R = \sum_{d,w} n(d,w)$$

Este algoritmo fue escogido ya que aporta la inclusión de medidas probabilísticas del modelo Topic Cube al modelo multidimensional propuesto, donde el estudio de sus resultados fueron buenos y permiten mayor rango de análisis de los datos no estructurados para el usuario final.

2.4 ALGORITMO IGBHSK

IGBHSK [8] es un algoritmo de agrupamiento no jerárquico de documentos híbrido entre el algoritmo de la mejor búsqueda armónica global y el algoritmo K-means. Este algoritmo necesita de una etapa inicial de pre-procesamiento de los documentos y finalmente una etapa de etiquetado, en la que se le asignan los nombres a los grupos de documentos.

IGBHSK utiliza el algoritmo GBHS como una estrategia global de búsqueda en el espacio de la solución completa, el algoritmo K-means como una estrategia local para improvisar soluciones; y el Criterio de Información Bayesiano (BIC) o el índice de Davies-Bouldin (DB) para encontrar el número de grupos automáticamente. En cuanto a estos índices se puede utilizar uno u otro. En IGBHSK, cada vector solución utilizado en el algoritmo GBHS tiene diferente número de grupos (centroides), y la función objetivo (basada en BIC o DB) del algoritmo GBHS depende de la ubicación de los centroides en cada vector solución y del número de centroides (valor de K).

IGBHSK tiene una rutina principal que ejecuta tres pasos básicos: Inicializar los parámetros del algoritmo, inicializar los mejores resultados de la memoria y llamar la rutina GBHSK en varias iteraciones y finalmente, retornar el mejor resultado. A continuación se presentan estos pasos en detalle.

1. **Inicializa los parámetros del algoritmo:** IGBHSK necesita unos parámetros específicos, el tamaño de los mejores resultados de la memoria (BMRS) y otros parámetros del algoritmo (HMS, HMCR, PAR y NI), Minimizando el criterio BIC o DB, llamado Función de Fitness

2. **Inicializar los mejores resultados de la memoria y llamar la rutina GBHSK:** La memoria de los mejores resultados (BMR) es una dirección de memoria donde los mejores vectores solución son almacenados (ver Figura 13). Cada fila en BMR almacena el resultado de un llamado a la rutina GBHSK, en un ciclo básico. Cada vector fila en BMR tiene dos partes: los centroides y el valor de fitness del vector.

$$BMR = \begin{bmatrix} Centroids_1 & Fitness_1 \\ Centroids_2 & Fitness_2 \\ \vdots & \vdots \\ Centroids_{BMRS-1} & Fitness_{BMRS-1} \\ Centroids_{BMRS} & Fitness_{BMRS} \end{bmatrix}$$

Figura 13. Mejores resultados de la memoria, figura tomada de [8]

3. **Seleccionar los mejores resultados:** Encontrar y seleccionar los mejores resultados de la memoria de los mejores resultados (BMR). El mejor resultado es la fila con el valor de fitness más alto (minimizar f(x)). Esta fila es la mejor solución de agrupamiento (centroides y fitness).

4. **Asignar etiquetas a los grupos:** el algoritmo IGBHSK contempla dos métodos de asignación de etiquetas a cada grupo. El primero es un conjunto de términos estadísticamente más representativos (SRT) basado en el concepto probabilístico introducido por Smith y Medin [39] y el segundo es similar a Lingo [40], basado en frases frecuentes (FPH). Más adelante se presentan en detalle estos métodos.

5. **Solapar los grupos:** Finalmente, cada grupo incluye documentos que quedan también en otros grupos, si estos documentos están a una distancia menor o igual que la distancia promedio del grupo.

Estos pasos se resumen en el algoritmo de la Figura 14.

```
01  Eliminación de palabras vacías
02  Algoritmo de Lematización de Porter
03  Construcción de la matriz de Términos por Documentos o la
    matriz de Términos Frecuentes por Documentos
04   Eliminar las dimensiones con un rango igual a cero
05  Para cada i € [1, BMRS] hacer
06      BMR[i] = GBHSK (TDM o FTDM)
07  Continuar_Para
08  Seleccionar los mejores resultados
09  Asignar etiquetas a los grupos
10  Solapar los grupos
```

Figura 14. Inicializar los mejores resultados de la memoria y llamar la rutina GBHSK, figura tomada de [8]

IGBHSK puede ser ejecutado en diferentes formas (ver Figura 15). La primera opción que se puede escoger está relacionada con el modelo de representación de documentos (TDM o FTDM). La segunda opción consiste en seleccionar la función de fitness (BIC o DB), y finalmente, es necesario escoger el método de etiquetado (términos estadísticamente más representativos o frases frecuentes).

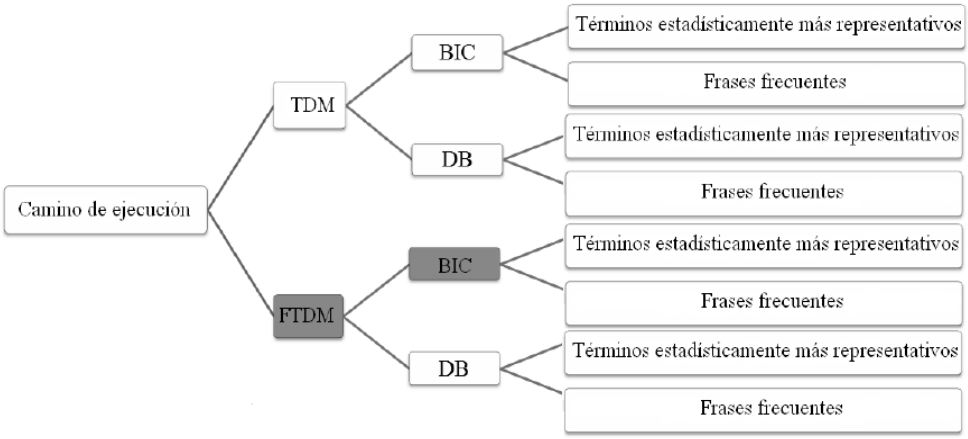


Figura 15. Caminos de ejecución del algoritmo IGBHSK, figura tomada de [8]

Los pasos del algoritmo IGBHSK se describen a continuación:

1. **Inicializar la memoria armónica:** La HM es una dirección de memoria donde todos los vectores solución son almacenados. Cada vector solución es creado con un numero aleatorio de centroides (k centroides con la estrategia de Forgy y valores en todas las dimensiones) y un fitness para esta solución. Los centroides iniciales son seleccionados aleatoriamente del conjunto de datos original (a diferencia del algoritmo original GBHS). La estructura general de HM es similar a BMR. En este paso, HMS vectores solución (centroides) son generados y luego el valor de fitness para cada vector es calculado.

2. **Improvisar una nueva armonía:** un nuevo vector armónico (centroides) es generado. Una variación del paso 3 en el algoritmo original GBHS se usa para crear los centroides en la solución actual. El proceso de selección aleatoria se ejecuta desde el conjunto de datos original (estrategia de Forgy). Después, un ciclo del algoritmo k-means se ejecuta y luego el valor de fitness para esta solución se calcula.

3. **Actualizar la memoria armónica:** el nuevo vector armónico reemplaza el peor vector armónico en la HM, si su valor de fitness es mejor que el segundo.

4. **Verificar el criterio de parada:** si el número máximo de improvisaciones (NI) se satisface, la iteración termina. De otro modo, los pasos 2 y 3 en GBHSK se repiten.

5. **Seleccionar la mejor armonía en HM:** la mejor armonía, que tiene el mínimo valor de fitness, es encontrada y seleccionada. Luego, el algoritmo k-means (Figura 7 sin el paso 01, porque tiene información sobre los centroides iniciales) se ejecuta y después se calcula un nuevo valor de fitness con la ubicación final de los centroides.

6. **Retornar el mejor resultado en la memoria armónica:** Retornar la mejor armonía (centroides y fitness) para IGBHSK.

Teniendo en cuenta que el algoritmo IGBHSK permite el agrupamiento de documentos a un solo nivel y el etiquetado de cada grupo, fue necesario modificarlo para que permitiera una jerarquía multinivel y aportar la estructura jerárquica que se definió en el modelo

multidimensional, para este caso una jerarquía de tópicos que trata un conjunto de documentos.

2.5 ARQUITECTURA DE UN SISTEMA

La arquitectura de un sistema es la disposición conjunta y ordenada de elementos software y hardware para cumplir una determinada función. No es difícil entender que si se mezclan arquitecturas distintas e inconsistentes sin ningún tipo de orden o planificación el proyecto puede convertirse fácilmente en incontrolable [41].

Dentro de un sistema de información, se definen ciertas configuraciones y procesos a partir de la lógica del negocio que se tenga, una arquitectura debería describir estas configuraciones y el entorno que facilite a los procesos crear nuevas funcionalidades que encajen en ella, incluyendo directivas, componentes de software, herramientas, etc.

Un sistema de información se define como un conjunto de componentes (o elementos) que operan conjuntamente para capturar, procesar, almacenar y distribuir información. Esta información se utiliza generalmente para la toma de decisiones, la coordinación, el control y el análisis en una organización. En muchas ocasiones la gestión de dicha información es el objetivo primario del sistema.

Al tener en cuenta como se constituye o se compone un sistema, a nivel de sus procesos, permite determinar el papel que desempeña la arquitectura dentro de los sistemas de información. Estos sistemas pueden modelarse según el estilo de arquitectura de niveles de abstracción o capas [42]. Este estilo se organiza jerárquicamente. Cada capa brinda servicios o datos a la capa siguiente.

En toda arquitectura de capa los elementos agrupados en una misma capa pueden comunicarse entre sí; pero existen variantes en cuanto a las comunicaciones permitidas entre elementos de capas diferentes:

- **Arquitectura top-down de capas:** Los elementos de una capa i pueden enviar solicitudes de servicio o datos a elementos de la capa inferior i+1. Típicamente se produce una cascada de solicitudes, es decir para satisfacer una solicitud de servicio o datos a una capa i+2, ésta requiere recibir solicitudes de servicio o datos de la capa i+1; cada una de estas solicitudes o datos de la capa i genera a su vez un conjunto de solicitudes o datos a la capa i+1 y así sucesivamente. Una arquitectura top-down es laxa (o no estricta) los elementos de una capa pueden enviar solicitudes de servicio o datos directamente a un elemento de cualquiera de las capas inferiores.

- **Arquitectura bottom-up de capas:** Cada elemento de una capa i puede notificar a elementos de la capa superior i+1 de que ha ocurrido algún evento de interés (ej. manejadores de dispositivos). La capa i+1 puede juntar varios eventos antes de notificar a su vez un elemento de la capa i+2. Una arquitectura bottom-up también puede ser no estricta si el elemento de la capa i puede notificar a cualquier elemento de cualquier capa superior a la capa i.

- **Arquitectura bidireccional de capas:** En su forma más común involucra dos pilas de N capas que se comunican entre sí. El ejemplo más conocido es el de los protocolos en Redes de Computadores.

De acuerdo a esto y teniendo en cuenta que al incorporar datos no estructurados en una bodega de documentos requiere de procesos que interactúan entre sí, se definió una arquitectura que permitiera relacionar estos procesos y proporcionara una idea general al incorporar este tipo de datos en una bodega de documentos.

3. MODELO PROPUESTO DE BODEGA DE DOCUMENTOS

3.1 ARQUITECTURA GENERAL

Con base en las características más importantes de Topic Cube y Text-Cube, el modelo inicial propuesto, presentadas en la Tabla 1, incorpora características de cada uno de los modelos en mención, para solucionar las diferentes restricciones presentadas, obteniendo un modelo hibrido, el cual a través de los diferentes procesos integra datos no estructurados en una bodega de documentos.

	Tipo de Medida	**Granularidad**	**Jerarquía**	**Dimensión Tópico**
Text Cube	Medidas Estándar de Recuperación de la Información (TF-IDF). Estas medidas no proporcionan un buen análisis ya que está basado en técnicas de conteo.	Documento. Esta granularidad proporciona en gran detalle información importante del documento.	Términos generada a través del documento. Esta no permite obtener información de los temas o sobre que trata un documento.	No. Al no poseer una dimensión tópico no genera ni registra una jerarquía de tópicos.
Topic Cube	Medidas de Recuperación de la Información Probabilísticas por medio del algoritmo PLSA. Este tipo de medidas permiten un mayor rango de análisis para el usuario final.	Tópico. Este tipo de granularidad omite información detallada e importante del documento.	Tópicos generada por un experto.	Sí. Al definir una dimensión tópico permite registrar la jerarquía de tópicos generada por el experto.

Modelo Inicial Propuesto	Medidas de Recuperación de la Información Probabilísticas por medio del algoritmo PLSA e inclusión de estas medidas probabilísticas a las medidas textuales. Este tipo de medidas junto con las medidas textuales permiten un mayor rango de análisis para el usuario final.	Documento. Esta granularidad proporciona en gran detalle información importante del documento.	Tópicos. Generados automáticamente por medio del algoritmo IGBHSK modificado.	Sí. Al definir una dimensión tópico permite almacenar la jerarquía de tópicos generada automáticamente por el algoritmo IGBHSK modificado.

Tabla 1. Tabla comparativa modelos Text-Cube, Topic-Cube y Modelo Inicial Propuesto

Para integrar datos no estructurados en una bodega de documentos, se definió una arquitectura general con base en el modelo inicial propuesto (ver Figura 16). Esta arquitectura está compuesta por cuatro procesos principales que permiten integrar los documentos al modelo, cada proceso recibe como entrada datos o documentos que se manipularon, para luego generar otros datos o documentos que serán las entradas de los siguientes procesos (arquitectura top-down). Estos procesos son: Generación de jerarquía, Medidas probabilísticas, proceso ETL y Cubo multidimensional.

La finalidad de la arquitectura es la integración de los datos no estructurados en una bodega de documentos a través los procesos antes mencionados. Esta arquitectura se definió durante todo el desarrollo del proyecto, ya que en ella se plasma, los cuatro procesos principales con funcionalidades particulares pero que tienen un objetivo igual que cumplir (agrupar, procesar, crear medidas, almacenar, mostrar información, entre otros).

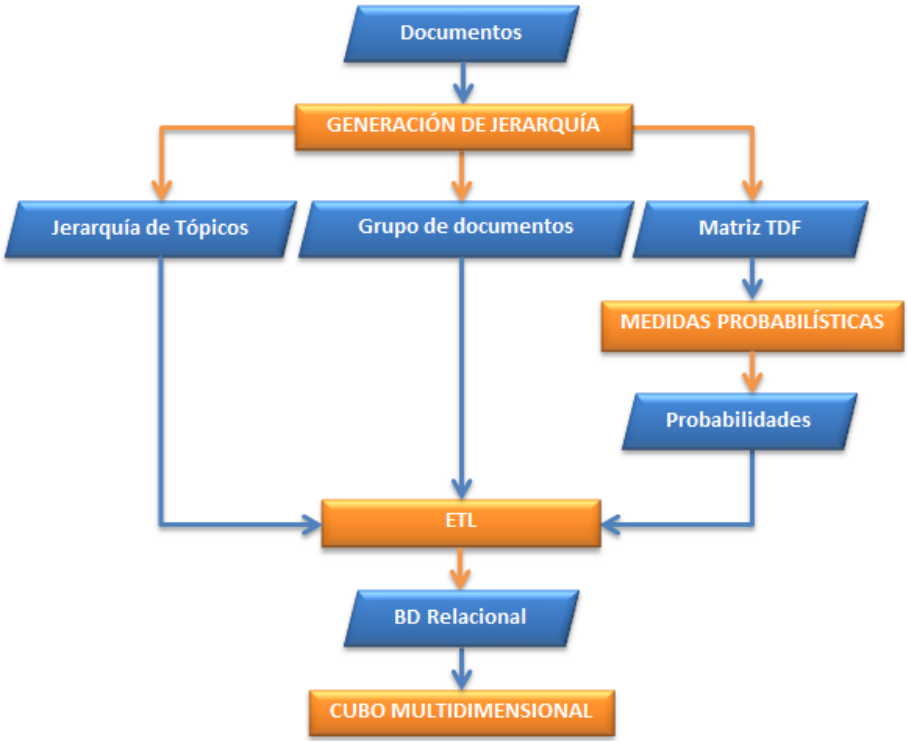


Figura 16. Arquitectura general propuesta

El primer proceso de la arquitectura, ***Generación de jerarquía*** (ver Figura 17), recibe el conjunto de documentos que requieren ser procesados para obtener los datos que serán ingresados en el modelo multidimensional. Estos documentos son procesados por dos algoritmos: Cosme (Paso 1) e IGBHSK [8] modificado. Cosme es un algoritmo que se desarrolló en este proyecto y está divido en dos pasos, el primero (Cosme - paso 1) para permitir: (1) la Conversión de documentos PDF a archivos de texto (.TXT), (2) Limpieza de los archivos de texto, extraer caracteres extraños resultantes de la conversión, (3) Generación de una serie de archivos XML (Conjunto de datos) que condensan todos los datos relevantes de los documentos inicialmente ingresados. El segundo paso de este algoritmo (Cosme - paso 2) consiste en el cruce de todos los archivos XML generados junto con las salidas del algoritmo IGBHSK modificado, estos forman nuevos archivos XML que representan los datos a cargar en el proceso ETL que más adelante se define para el modelo planteado. El algoritmo IGBHSK por su parte, fue modificado para que recibiera como entrada los archivos entregados por el algoritmo Cosme y adicionalmente para generar una jerarquía multinivel. IGBHSK modificado realiza un pre-procesamiento a los archivos de entrada específicamente el texto del documento a procesar, construye la matriz de términos por documentos (TDM) basado en técnicas de conteo como son TDF-IDF, realiza un agrupamiento de los documentos que son utilizados para la generación de la jerarquía multinivel junto con su respectivo etiquetado. Cada iteración de este algoritmo genera un nuevo conjunto de documentos del siguiente grupo a analizar y de esta manera se va construyendo la jerarquía multinivel. Una vez terminado el proceso iterativo de la creación de la jerarquía, éste algoritmo produce las siguientes salidas necesarias para los siguientes procesos en nuestra arquitectura: una jerarquía de tópicos con sus respectivas etiquetas, archivos XML con los documentos clasificados y la matriz TDF. Es preciso

aclarar dos cosas: primero, si los archivos originales están en un formato diferente de texto, estos deben ser convertidos con alguna herramienta, tal como se realiza con los archivos PDF y pasar directamente al algoritmo IGBHSK modificado, saltándose la ejecución del paso uno del algoritmo Cosme. Segundo, la jerarquía de tópicos no necesariamente tiene un nivel fijo en su estructura ya que el nivel de profundidad depende del número de documentos procesados.

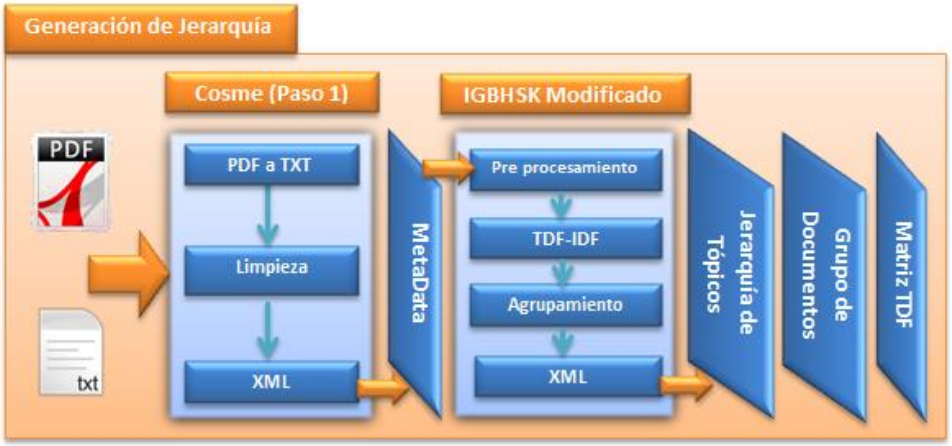


Figura 17. Generación de Jerarquía

Un segundo proceso, ***Medidas probabilísticas*** (ver Figura 18), se encarga de la ejecución del algoritmo PLSA [9], recibiendo como entrada la matriz de términos por documento y el número de tópicos generados por IGBHSK modificado. Al aplicar el modelo PLSA a un conjunto de documentos representados en la matriz de términos, este normaliza la matriz y extrae un conjunto de probabilidades que caracterizan a cada tópico en la jerarquía (para nuestro caso, los tópicos del último nivel de la jerarquía), es decir, este proceso permite asignar una probabilidad a cada documento de acuerdo al tópico en el que se asigna, y adicionalmente, obtener y asignar las palabras con sus probabilidades a cada tópico. Con base en lo anterior, PLSA utiliza el algoritmo EM [38, 43] para mejorar las probabilidades de los tópicos en los documentos $P(z|d)$ y de las probabilidades de las palabras en los tópicos $P(w|z)$ de acuerdo a una serie de iteraciones. Como salida se obtienen los datos probabilísticos en archivos TXT que requiere el siguiente proceso.

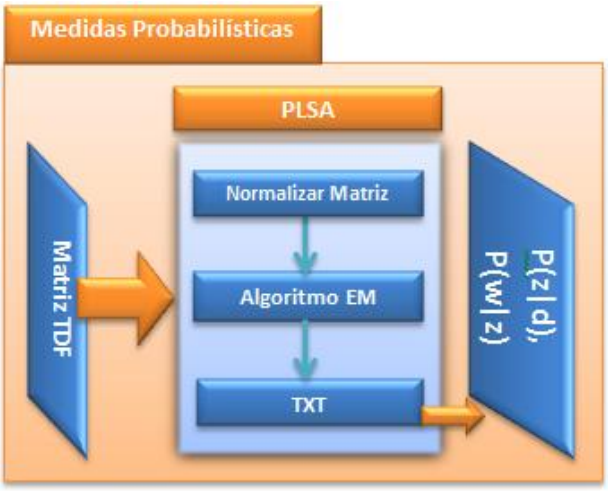


Figura 18. Medidas Probabilísticas

Un tercer proceso de la arquitectura es el ***ETL*** (Proceso de Extracción, Transformación y Carga, por sus siglas en ingles *Extract-Transform-Load*) [12], ver Figura 19, hace referencia al cargue de los datos en una base de datos relacional. Inicialmente el algoritmo Cosme (Paso 2) recibe un conjunto de archivos XML que contienen toda la información entregada por los procesos anteriores, los cuales se cruzan entre sí para generar información que cumple con la estructura definida en el modelo multidimensional y de esta forma poder realizar el cargue de una forma directa. Este cargue se realiza por medio de una de las herramienta ETL existentes, de tal forma que los datos queden dentro de la base de datos relacional de la bodega. En el proceso ETL se realizan tres subprocesos principales, *Extracción:* encargado de recuperar los datos de las fuentes de información para nuestro caso archivos XML; *Transformación:* aplica una serie de reglas de negocio y/o funciones de tipado sobre los datos extraídos para convertirlos en datos que serán cargados; *Carga:* es el momento en el cual los datos transformados son cargados en la base de datos relacional del modelo, en esta operación se aplican las restricciones que se hayan definido (por ejemplo, valores únicos, integridad referencial, campos obligatorios, rangos de valores entre otros) las cuales garantizan la calidad de los datos en el proceso ETL. Al final de este proceso de ETL se obtiene una base de datos relacional con los datos cargados y lista para el proceso de creación del Cubo multidimensional.

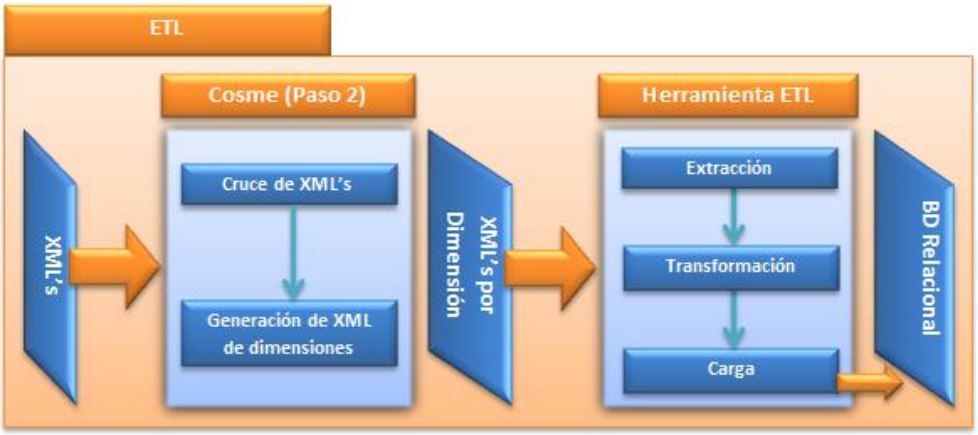


Figura 19. ETL – Extracción, Transformación y Carga

El cuarto y último proceso es la creación del ***Cubo multidimensional*** (ver Figura 20). Una bodega de datos (En inglés, DataWarehouse) [12] almacena datos en diversas dimensiones que conforman un cubo multidimensional, en donde el cruce de los valores de cada dimensión, determinan un hecho en específico a través de las diferentes medidas creadas, para este caso, medidas de texto. El objetivo es conseguir una mayor eficiencia y desempeño en las consultas. Por su parte un cubo multidimensional provee una estructura que permite tener acceso flexible a los datos para explorar, analizar sus relaciones, y utilizar la información desde diferentes perspectivas de análisis de la información presentada, en nuestro caso, la información a partir de medidas textuales de los diferentes documentos en mención. El uso de una herramienta OLAP o de un generador de reportes adecuado, permite a los usuarios visualizar y analizar la información de una forma natural e intuitiva generando consultas multidimensionales, con columnas y filas móviles. Los subprocesos que intervienen en el cubo dimensional son: la generación de las medidas de texto escritas en lenguaje MDX (Lenguaje que se ha convertido en una herramienta poderosa que permite realizar consultas multidimensionales avanzadas y complejas sobre un cubo OLAP); luego la generación y ejecución de la función de agregación, que se realizó por medio de un ensamblado. En el capítulo 3.4 se presentan detalles relacionados con las medidas textuales y las funciones de agregación. Y por último el procesamiento del cubo multidimensional para dejarlo listo para las consultas del usuario final.

Figura 20. Cubo Multidimensional

3.2 MODELO MULTIDIMENSIONAL PARA UNA BODEGA DE DOCUMENTOS

El modelo multidimensional propuesto (ver Figura 21), se presenta como un esquema estrella (el cual se explicó en el capítulo 2.1.2), el cual presenta una tabla de hecho principal llamada *FactAssignation*, con una granularidad a nivel de documento, con dimensiones estándar, una dimensión tópico, relaciones Muchos-a-Muchos entre la tabla de hechos principal y algunas dimensiones, y medidas textuales con los valores de las probabilidades obtenidos con el algoritmo PLSA.

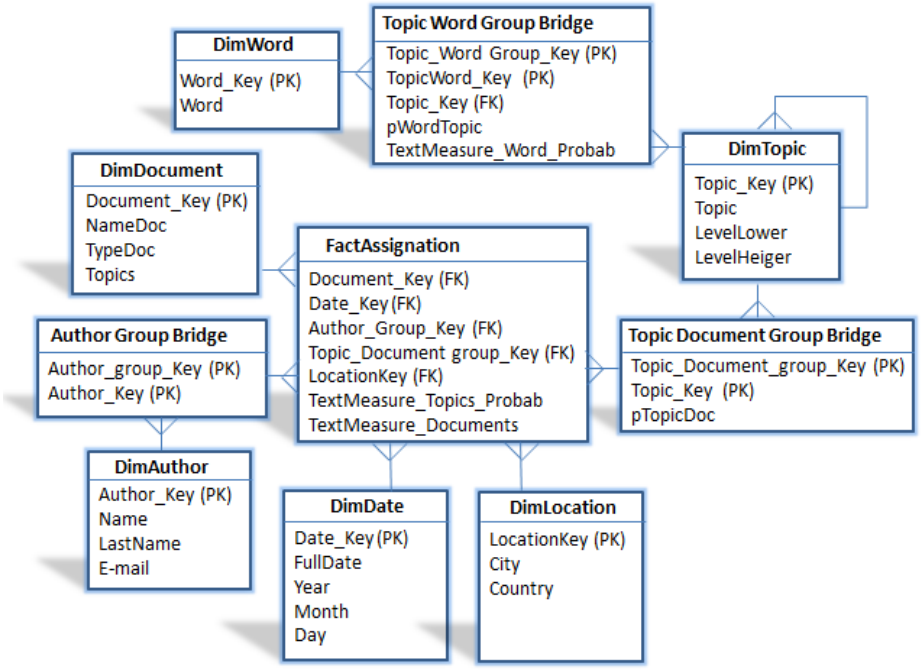


Figura 21. Modelo Multidimensional

El modelo se compone de cuatro partes: la dimensión *Topic*, la dimensión *Document*, la dimensión *Word* y por último las dos medidas textuales agregadas (*TextMeasure_Topics_Probab* y *TextMeasure_Documents*). Estas medidas textuales pueden utilizarse para analizar información correspondiente a documentos de texto, pero para suplir más necesidades de los usuarios, se pueden agregar nuevas medidas al modelo propuesto.

3.2.1 Dimensiones estándar

En el modelo se presentan dimensiones relacionadas con información relevante del documento, como: la dimensión *documento* con el título, el tipo de documento; *autor* con los nombres y el *e-mail* del autor; *fecha* con la fecha de la publicación que permite hacer los análisis tradicionales de fechas; *palabra* con todas las palabras del conjunto de documentos definidos en los tópicos almacenados en la bodega.

3.2.2 Dimensión tópico

La dimensión *tópico* es especial, ya que permite almacenar la jerarquía de tópicos definida para el conjunto de documentos que serán almacenados en la bodega de documentos. Esta jerarquía se modeló como una jerarquía padre-hijo[15] la cual permite tener un árbol jerárquico balanceado con los tópicos que se definan automáticamente. El último nivel de la jerarquía de tópicos se relaciona con el documento, de acuerdo al contenido del mismo. Esta jerarquía se genera automáticamente por medio del algoritmo IGBHSK modificado, explicado más adelante en el capítulo 4.1.2.

3.2.3 Relaciones muchos-a-muchos

El modelo presenta una relación Muchos-a-Muchos entre la tabla de hechos principal (*FactAssignation*) y la dimensión autor (*DimAuthor)* a través de una tabla de puente (*Author Group Bridge*), que permite almacenar los autores de un documento determinado, debido a que los documentos pueden ser creados por más de un autor (especialmente para el caso de artículos científicos). También se presenta una relación Muchos-a-Muchos entre la tabla de hechos principal y la dimensión tópico (*DimTopic*) a través de la tabla puente (*Topic Document Group Bridge),* debido a que un documento puede estar relacionado con más de un tópico, para ello se almacena una medida numérica con la probabilidad del documento en un tópico específico (*pTopicDoc*). Por último se modela una relación Muchos-a-Muchos entre las dimensiones tópico y la dimensión palabra (*DimWord*) a través de la tabla puente (*Topic Word Group Bridge*), para almacenar una medida numérica con la probabilidad de que una palabra de un documento este en un tópico en específico (*pWordTopic*).

3.2.4 Medidas de la tabla de hechos principal y la tabla puente de las dimensiones tópico y palabra

La tabla de hechos principal presenta una medida textual calculada (*TextMeasure_Topics_Probab*), la cual representa las probabilidades de un documento con respecto a los tópicos principales que trata el documento. Esta medida, le permite al usuario final identificar cuáles son los principales tópicos contenidos en los documentos y adicionalmente visualizar el porcentaje que representa para cada tópico. El modelo contempla otra medida textual calculada (TextMeasure_Documents), la cual permite obtener los documentos con sus porcentajes de acuerdo a las dimensiones que se relacionan en una consulta, como por ejemplo, obtener los documentos por tópico, los documentos por autor, o los documentos por fecha.

La tabla puente entre las dimensiones tópico y palabra, presenta otra medida textual calculada que presenta las probabilidades de una palabra en los principales tópicos (*TextMeasure_Word_Probab*), está por su parte, permite que se identifiquen cuáles son las palabras más relevantes en un tópico específico con el porcentaje que representa en

[15] El modelo jerárquico facilita relaciones 1:N (uno a muchos), de manera que un padre puede tener más de un hijo, todos ellos localizados en el mismo nivel, y un hijo solo puede tener un padre situado en el nivel inmediatamente superior al suyo.

cada tópico. Este tipo de medidas permiten realizar un análisis tanto cualitativo como cuantitativo, mejorando la toma de decisiones a nivel de una organización así como de un usuario final.

3.3 MODELO LÓGICO

Se elaboró el modelo lógico de la estructura de la bodega, con base en el modelo multidimensional propuesto en el capítulo anterior, que soporta las consultas a través de las medidas textuales, las cuales arrojaron información correspondiente a los documentos procesados a través de las diferentes dimensionalidades. Se puede observar en la Figura 22 el mapeo a un modelo lógico con respecto a las particularidades de la implementación en la herramienta Microsoft Analysis Services 2008 (Anexo A), esta herramienta permite diseñar, crear y administrar estructuras multidimensionales que contienen detalles y datos agregados de diferentes fuentes de información. Para el caso de estudio, permitió definir tanto las medidas textuales como la función de agregación, que realizan las diferentes operaciones típicas OLAP.

La base de datos relacional correspondiente al modelo multidimensional propuesto, se diseñó e implementó en el motor de base de datos de Microsoft SQL server 2008, el cual permite almacenar la información correspondiente a cada tabla de hechos y sus dimensiones. La creación del cubo OLAP de ésta base de datos se realizó por medio de la herramienta SQL Server Analysis Services 2008 (Anexo B). Para el análisis del cubo multidimensional por parte del usuario final se requiere de una herramienta OLAP que permita interactuar con las dimensiones y medidas, para este proyecto se utilizó la herramienta de evaluación Dundas OLAP, que dispone de un conjunto de funciones avanzadas para ASP.net que permitieron la interacción en línea con el cubo.

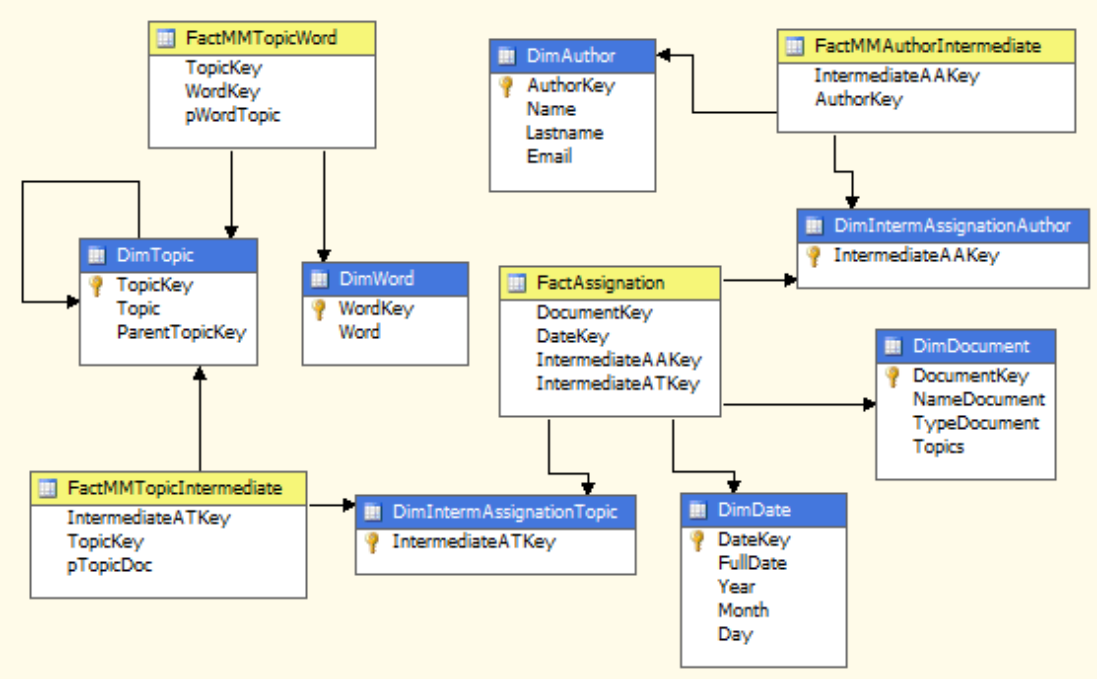


Figura 22. Modelo Lógico

La forma de implementación del modelo está muy sujeto al entorno de desarrollo que se provee para tal fin, por tal razón debe tenerse en cuenta cómo mapear las diferentes relaciones y características del modelo en el entorno de desarrollo de elección. Características como las relaciones muchos a muchos entre la tabla de hechos *FactAssiganation* con las dimensiones *Author, Topic y Word*, también la relación recursiva en la dimensión *Topic*.

3.4 MEDIDAS TEXTUALES Y FUNCIÓN DE AGREGACIÓN

3.4.1 MDX

Como se definió en el modelo multidimensional, las medidas: *TextMeasure_Topic_Probab, TextMeasure_Documents* y *TextMeasure_Word_Probab*, son textuales; para poder manipularlas se crean como medidas textuales calculadas mediante MDX (por sus siglas en ingles *MultiDimensional eXpressions*) [44]. Este lenguaje es usado para interactuar con los mecanismos de almacenamiento y de recuperación de datos en las bases multidimensionales sobre cubos OLAP (ver ejemplo 1) diferentes al de la tecnología de las bases de datos relacionales ya que tiene funciones y fórmulas que lo hacen muy potente para el análisis de los datos. Este mecanismo se basa en el concepto de espacio n-dimensional[16]. En las bases de datos relacionales el lenguaje SQL es usado para ensamblar un conjunto de datos, con Analysis Services el lenguaje MDX es usado para ensamblar tuplas[17] para identificar puntos de los datos dentro del espacio n-dimensional [45]. En resumen es el lenguaje a través del cual se puede explotar la información que reside en los motores OLAP y satisfacer las consultas analíticas que se realizan.

Ejemplo 1.

```
SELECT
    [Medida] on 0,
    [Dimension].[Jerarquia].[Nivel] on 1
FROM [Nombre-de-la-bodega-de-datos]
```

Las expresiones son una parte crítica del lenguaje y son unidades de código MDX que pueden ser evaluadas para retornar un valor u objeto de referencia. Adicionalmente, las expresiones MDX pueden ser usadas para adicionar lógica del negocio en los cubos, configuración simple o avanzada de seguridad, códigos de colores para propósitos de alerta en las excepciones, miembros roll-ups personalizados, niveles roll-ups personalizados, acciones, entre otros. En otras palabras MDX es usado en casi todas las partes del diseño efectivo de cubos OLAP.

MDX devuelve un conjunto de celdas resultado de tomar un subconjunto de las celdas del cubo original a través de sus múltiples funciones y fórmulas que lo hacen muy potente para el análisis de los datos, esto es porque las estructuras dimensionales están

[16] Es el espacio de referencia donde cada eje de coordenadas representa una dimensión con respecto a un modelo multidimensional.
[17] Una tupla es una colección de coordenadas que referencia la intersección de uno o más miembros de cada una de las dimensiones en el cubo.

jerarquizadas y se representan en forma de árbol y por lo tanto existen relaciones entre los diferentes miembros de las dimensiones, permitiendo referenciar los diferentes elementos de las dimensiones con expresiones como *Miembros-Hijo*, *Miembros-Primo*, *Miembros-Padre*, etc. haciendo una analogía con un árbol genealógico.

Como antes se había mencionado, MDX permite implementar acciones personalizadas de acuerdo a la lógica del negocio o necesidades para recuperar información que no se establece directamente en el modelo multidimensional, particularmente, una medida calculada se crea a partir de una expresión MDX que se define y se calcula en tiempo de ejecución. Esta medida calculada no tiene datos almacenados en los puntos del espacio del cubo asociado a las dimensiones referenciadas, lo que permite evaluar, en cada punto o celda asociado a los miembros de las dimensiones calculadas, una expresión (suma, resta, promedio, concatenar, etc.) que arroja un valor calculado para ser visualizado. Utilizar estas medidas calculadas en una consulta permite manipular los datos, en nuestro caso, manipular una medida textual a partir de funciones MDX para facilitar el análisis de los hechos.

3.4.2 Función de agregación

Toda medida tiene asociada una función de agregación que por defecto es la suma, donde se resume algún grupo de columnas retornando el valor de cada grupo al subir (*roll-up*) o al bajar (*drill-down*) por la estructura dimensional, pre calculando el valor de los datos y disminuyendo el tiempo de respuesta en la consulta. Esta función de agregación se puede personalizar o crear para que cumpla con algunos requerimientos especiales, para este caso, promediar los tópicos que se repitan en una consulta mediante un Procedimiento Almacenado, donde se reciben los tópicos con sus probabilidades en formato de texto generado a partir de una medida calculada mediante el lenguaje de consulta MDX, manipular y encontrar cuales son los tópicos que se repiten y promediar sus probabilidades. Por otro lado se personaliza una función de agregación en una expresión MDX que permita promediar las probabilidades de las palabras que se repitan en una consulta.

En el modelo propuesto, un *procedimiento almacenado*[46]es utilizado para llamar rutinas externas a la herramienta de implementación. Estas rutinas son implementadas en forma de ensamblados en cualquier lenguaje compatible con MSIL (Microsoft Common Intermediate Language) tales como *C, C++, C#, Visual Basic*, entre otros. Un *procedimiento almacenado* se crea una vez y puede ser llamado desde muchos contextos, en especial desde una medida calculada permitiendo una mayor funcionalidad de la medida textual que se desea implementar debido a que este extiende las funciones nativas de MDX.

3.4.3 Generación de medidas textuales

Para poder utilizar *MDX* y un *procedimiento almacenado* que defina la función de agregación, se necesita generar tres medidas textuales calculadas que permitan manipular los datos de tipo texto y realizar las operaciones típicas OLAP. Una medida que

permita visualizar los tópicos con sus probabilidades de acuerdo a las dimensiones estándar definidas en el modelo; una medida que permita visualizar las palabras con sus probabilidades de acuerdo a la dimensión tópico definida en el modelo; y otra medida que permita visualizar el nombre de los documentos de acuerdo a las dimensiones estándar relacionadas en una consulta. Para definir estas tres medidas calculadas, se deben realizan los siguientes pasos:

i) Medida *TextMeasure_Topics_Probab* (ver Figura 21) (Anexo C):

1. Se crea una medida de tipo numérico, *Measure_DocumentKey* a partir de la llave foránea de la dimensión tópico en la tabla de hechos.
2. Se crea una medida textual *MA*, *TextMeasure_Topics*, que relaciona la medida del paso 1 con el atributo *Topics* de la dimensión *documento* que retornará los tópicos con sus probabilidades de cada documento.
3. Se crea una medida textual, *TextMeasure_Topics_Probab*, que relacione la medida del paso 2 con las dimensiones estándar (*DimAuthor, DimDate, etc.*) y permita mediante una función de MDX concatenar, de acuerdo al tipo de consulta, los tópicos con sus probabilidades. La función MDX retorna el valor del texto de acuerdo a las dimensiones relacionadas y jerarquías definidas, luego recorre los elementos de la dimensión documento, si el valor de la medida 2 existe se concatena el valor, sino lo omite.
4. Se define un *procedimiento almacenado* (Anexo D), que recibe la medida del paso 3 y promedia las probabilidades de los tópicos que se repitan en una consulta dada, dándole un formato adecuado de presentación y así permitir que la función de agregación trabaje para los diferentes niveles de las estructuras dimensionales.

Para entender mejor el funcionamiento de esta medida textual, suponga que cada celda en un cubo *C*, dependiendo de las dimensiones relacionadas, representa una medida textual *MT* creada con base en la medida calculada del paso 2 *MA*, la cual contiene N tópicos con sus probabilidades *TPi* separados por punto y coma (;), ésta a su vez compuesta por el tópico *Ti* y su probabilidad *Pi* separados por dos puntos (:). El pseudocódigo del Cuadro 1 recibe la medida textual *MT,* compara los tópicos que se repiten y permite a la función de agregación promediar las probabilidades, retornando una nueva medida textual. Al definirse esta medida textual los tópicos son agregados a través de las diferentes dimensionalidades del cubo OLAP generado. En la Figura 23 se visualiza como la función de agregación calcula el promedio de la medida textual.

```
Para cada consulta en el cubo C
Iterar hasta el número de elementos de la consulta
    Si el valor de la medida MA no es null hacer
                    Si la medida MA tiene TP's
                            MT ←concatenar(MA_{i-1}, MA_i)
                    Fin si
    Fin si
Fin iteración

Para cada medida MT
Procedimiento StoredProcedure(MT)
    Para i←0 hasta el número de tópicos de MT hacer
        ListaTP[i] ←TP_i
    Fin Para
    Para i ← 0hasta longitud de ListaTPhacer
      ListaT[i] ← Ti
      ListaP[i] ← Pi
    Fin Para
    Para i←0 hasta el número de tópicos enListaT hacer
          tempT←ListaT[i]; tempP←ListaP[i]
          borrartempTde ListaT; borrartempP de ListaP
          Para j←0 hasta el número de tópicos enListaT hacer
              Si tempTes igual ListaT[j]
                      ListaPosicion←j
              Fin Si
          Fin Para
          Si longitud ListaPosicion es mayor que 0 hacer
            Paraj←0hasta la longitud ListaPosicion hacer
                      tempP←tempP+ListP[ListaPosicion[j]]
            Fin Para
            Para j←0 hasta longitud ListaPosicionhacer
                borrar de ListaT la posición ListaPosicion[j]
                borrar de ListaPla posición ListaPosicion[j]
            Fin Para
            tempP←(tempP/(longitud ListaPosicion))*100
            Concatenar(NuevoMT, tempT, tempP)
          Si No
            Concatenar(NuevoMT, tempT, tempP)
          Fin Si No
      Fin Para
      Retornar NuevoMT
Fin Procedimiento
```

Cuadro 1. Pseudocódigo medida textual *TextMeasure_Topics_Probab*

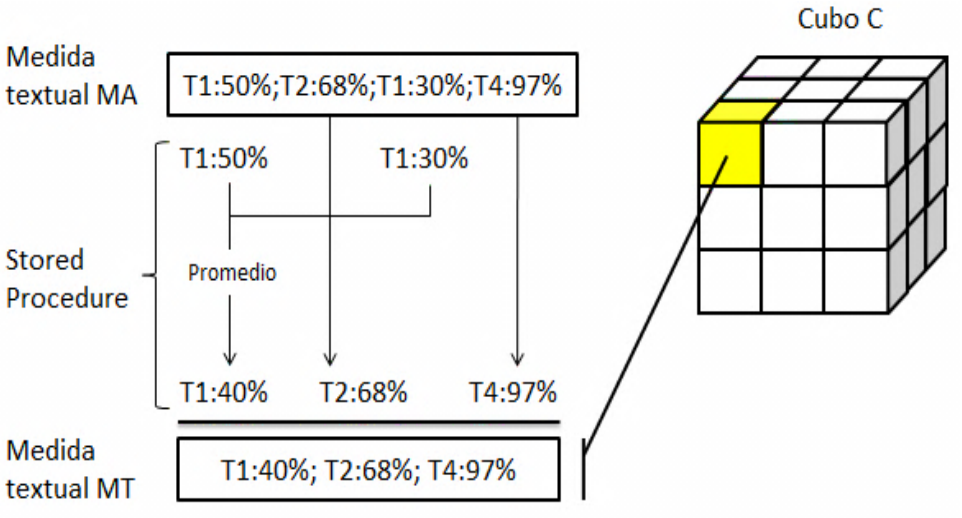


Figura 23. Medida textual *MT* tópicos

ii) Medida *TextMeasure_Word_Probab* (ver Figura **21**) (Anexo E):

1. Se crea una medida textual dentro de la tabla de hechos intermedia *FactMMTopicWord*. Como la relación de las dimensiones hacia la tabla de hechos intermedia es de *uno-a-muchos* no es necesario tener una referencia del atributo Word con la llave foránea de la dimensión *DimWord* ya que está directamente relacionada con la dimensión *DimTopics*. En este caso se recorre la *DimWord* mediante una función de MDX que permita concatenar las palabras que tiene un tópico, adicionalmente la herramienta de implementación genera una medida de recuento por defecto en la tabla de hechos intermedia, si el valor que toma la medida de recuento es diferente de cero (0), se extrae la palabra y se concatena con la probabilidad de esa palabra en cada tópico, de lo contrario se omita el valor. La función de agregación asociada a la medida textual es el promedio, como la medida recuento tiene asociada como función de agregación la suma, solo se necesita que en la expresión MDX se haga el correspondiente promedio, es decir tomar la probabilidad asociada a cada palabra, dividirla por la medida recuento y multiplicarla por cien.

Para entender mejor el funcionamiento de esta medida textual, suponga que cada celda en un cubo *C*, relacionando las dimensiones *DimTopic* y la *DimWord*, representa una medida textual *MT* la cual contienen *N* palabras con sus probabilidades separados por un punto y coma (;), ésta a su vez compuesta por la palabra *Wi* y su probabilidad *Pi* obtenida de *pWordTopic* separados por dos puntos (:). El pseudocódigo del Cuadro 2 construye la medida textual a partir de una función MDX. Al definirse esta medida textual las palabras o términos son agregados a través de las dimensiones *DimWord* y *DimTopics* del cubo OLAP generado. En la Figura 24 se visualiza, de forma general, como la medida textual trabaja con la función de agregación promedio.

```
Para la consulta en el cubo C
Iterar hasta el número de elementos de la DimWord
    Si el valor de la medida Recuento es diferente de cero hacer
        MT ← concatenar (Wi, (Pi/Recuento)*100)
    Fin si
Fin iteración
```

Cuadro 2. Pseudocódigo medida textual *TextMeasure_Word_Probab*

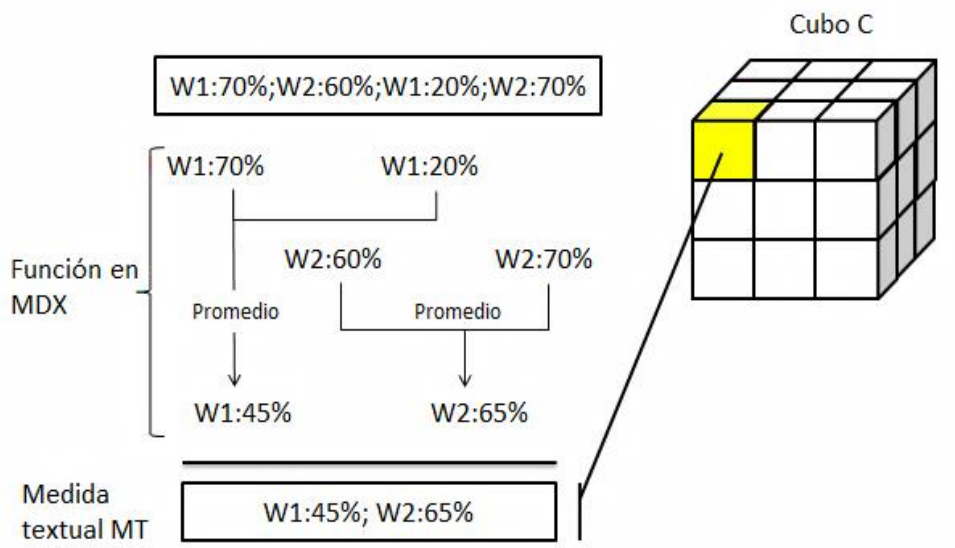


Figura 24. Medida textual MT palabras o términos

iii) Medida *TextMeasure_Documents* (ver Figura 21) (Anexo F):

1. Se crea una medida textual, *TextMeasure_Doc*, que se relaciona con el atributo *NameDocument* de la dimensión documento, el cual retornará los documentos con sus probabilidades de acuerdo al tipo de consulta.
2. Se crea una medida textual, *TextMeasure_Documents*, que relacione la medida del paso 1 con las dimensiones estándar (*DimAuthor, DimDate*, *etc*.) y permita mediante una función de MDX concatenar los documentos con sus probabilidades. La función MDX retorna el valor del texto de acuerdo a las dimensiones relacionadas y jerarquías definidas. Recorre los elementos de la dimensión documento, si el valor de la medida textual del paso 1 existe se concatena el valor, sino lo omite. Al tener la medida de recuento que genera por defecto la herramienta en la tabla de hechos intermedia *FactMMTopicIntermediate,* si el valor que toma la medida de recuento es diferente de cero (0), se extrae el nombre del documento y se concatena con su probabilidad, de lo contrario se omita el valor. La función de agregación asociada a la medida textual es el promedio, como la medida recuento tiene asociada como función de agregación la suma, solo se necesita que en la expresión MDX se haga el correspondiente promedio, es decir tomar la probabilidad asociada a cada documento, dividirla por la medida recuento y multiplicarla por cien.

Para entender mejor el funcionamiento de esta medida textual, suponga que cada celda en un cubo *C*, dependiendo de las dimensiones relacionadas, representa una medida textual *MT* creada con base en la medida calculada del paso 2 *MD*, la cual contienen *N* palabras con sus probabilidades *DPi* separados por un punto y coma (;), ésta a su vez compuesta por el documento *Di* y su probabilidad *Pi* obtenida de *pTopicDoc* separados por dos puntos (:). El pseudocódigo del Cuadro 3 construye la medida textual a partir de una función MDX y en la Figura 25 se visualiza, de forma general, como la medida textual trabaja con la función de agregación promedio.

```
Para cada consulta en el cubo C
Iterar hasta el número de elementos de la consulta
    Si el valor de la medida MD no es null hacer
                Si la medida MD tiene DP's
                        MT ← concatenar (Di, (Pi/Recuento)*100)
                Fin si
    Fin si
Fin iteración
```

Cuadro 3. Pseudocódigo medida textual *TextMeasure_Documents*

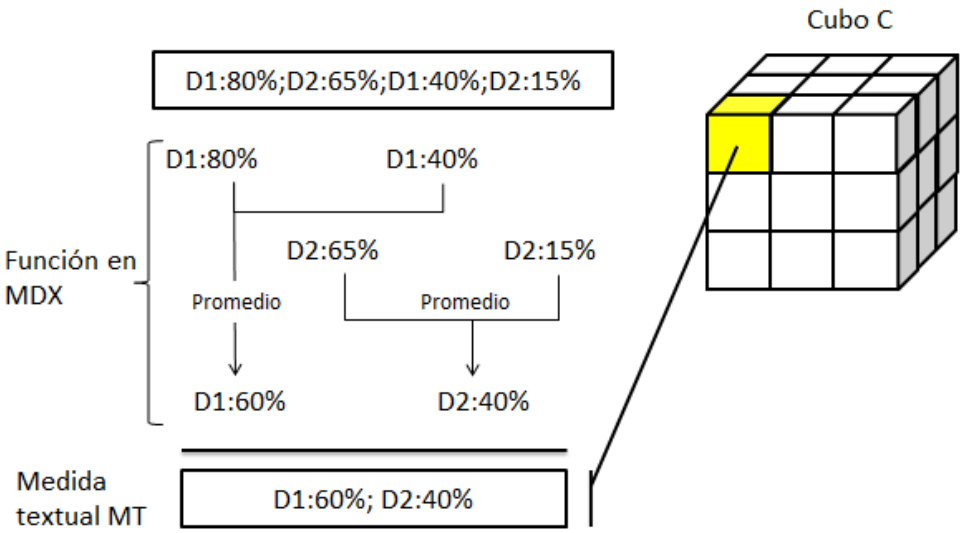


Figura 25. Medida textual MT documentos

Finalmente, al haber creado este tipo de medidas, el usuario final puede incorporar medidas textuales y numéricas en las consultas, que le permitan un análisis de los datos no estructurados en la toma de decisiones.

4. EVALUCION DEL MODELO MULTIDIMENSIONAL Y CONCLUSIONES

4.1 CARGUE DEL MODELO

Para realizar el cargue del modelo, se usó un conjunto de artículos científicos en formato pdf de la librería llamada EndNote usada por algunos profesores del departamento de sistemas de la Universidad del Cauca. Se siguieron los pasos definidos en la arquitectura del modelo para bodegas de documentos presentado en el capítulo 3.1. En resumen, los pasos son: *Pre-procesamiento de los documentos*, para hacer la remoción de caracteres especiales, paso a minúsculas, remoción de palabras vacías en inglés y stemming. Luego se construye la matriz de términos por documentos basado en la fórmula de Salton [47] (tf-idf); *definición de la jerarquía de tópicos*, parte de esta matriz para definir los grupos de documentos y colocarle la etiqueta a cada grupo; *Información básica del documento*, de cada documento se identifica información básica como: título y tipo del documento, nombres y correos electrónicos de los autores, y la fecha de publicación del mismo; *Probabilidades,* al tener la matriz de termino por documento, se calculan las probabilidades requeridas *pTopicDoc* y *pWordTopic* para definir las medidas calculadas antes mencionadas en el modelo multidimensional.

4.1.1 Pre procesamiento con algoritmo Cosme

El algoritmo Cosme (Anexo J) fue desarrollado dentro del proyecto en el lenguaje C# de Visual Studio .NET 2010 para apoyar el proceso de pre-procesamiento, cargue y transformación de los datos. Este está compuesto por dos pasos que se detallan a continuación:

Paso 1: este paso se encarga inicialmente de la conversión de archivos pdf a txt por medio de la librería llamada Neevia DocCreator[18] en su modo evaluación, debido a que fue la herramienta que más se adecuó a los requerimientos del proyecto, entre ellos, que realizara la conversión de más de 10 hojas, fácil acople con la tecnología .NET, sin procesar marcas de agua y procesar texto que se encuentra como imagen. Otras librerías como ItextSharp, Pdf to Text no tienen estas características de conversión. Seguidamente se realiza un proceso de extracción o remoción de caracteres extraños generados a partir del proceso conversión, por ultimo este paso realiza un generación de archivos XML correspondientes y/o usados como entradas en el algoritmo IGBHSK modificado; los archivos generados son: *docs_grupo_All_nivel_0.xml*, es el encargado de contener los meta datos de todos los documentos pre-procesados junto con el texto del documento original que será procesado por el algoritmo IGBHSK modificado para la generación de la jerarquía multinivel. *DimAuthor.xml*, *DimDate.xml*, *DimDocuments.xml*, *FactAssiganation* y *FactMMAuthor.xml* son archivos que contienen los datos a cargar en sus respectivas dimensiones en la bodega de documentos (ver Figura 26).

[18]http://www.neevia.com/products/cr/

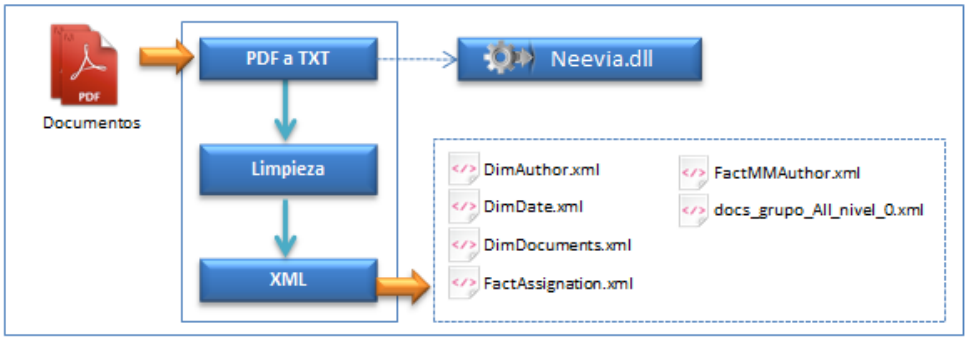


Figura 26. Paso 1 Algoritmo Cosme

Paso 2: Este paso inicialmente carga la jerarquía multinivel generada por IGBHSK modificado, los archivos de texto que contienen las probabilidades de los tópicos en los documentos *P(z|d)* y los archivos de texto que contienen la probabilidades de las palabras en los tópicos *P(w|z)*, seguidamente genera los archivos XML: DimTopic.xml, FactMMTopicIntermediate.xml, DimWord.xml, FactMMTopicWord.xml y actualiza el archivo DimDocuments.xml que van a ser cargados respectivamente en la bodega de datos (ver Figura 27).

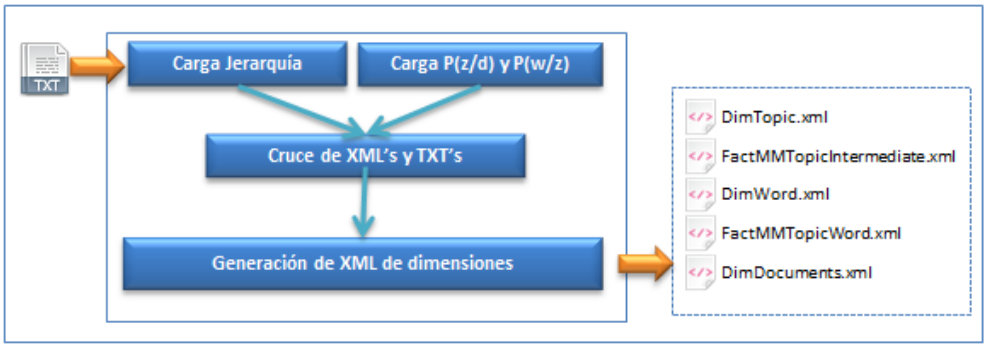


Figura 27. Paso 2 Algoritmo Cosme

4.1.2 Jerarquía de tópicos con IGBHSK modificado

El algoritmo IGBHSK original (ver capítulo 2.4) solo genera un nivel de la jerarquía, pero para la bodega se necesita una jerarquía multinivel que permita la navegación por esta jerarquía. Por esto fue necesario modificar el algoritmo IGBHSK (ver Figura 28), para obtener la jerarquía de documentos y asociar cada documento al último nivel de la jerarquía. Para lograr esto, se toma la base inicial del algoritmo IGBHSK y se realiza modificaciones para iterar sobre los procesos de creación de la matriz de términos por documento, agrupamiento y etiquetado, creando un nuevo conjunto de documentos por cada tópico generado, estos nuevos conjuntos de documentos son de nuevo procesados para ir formando la jerarquía tópicos multinivel.

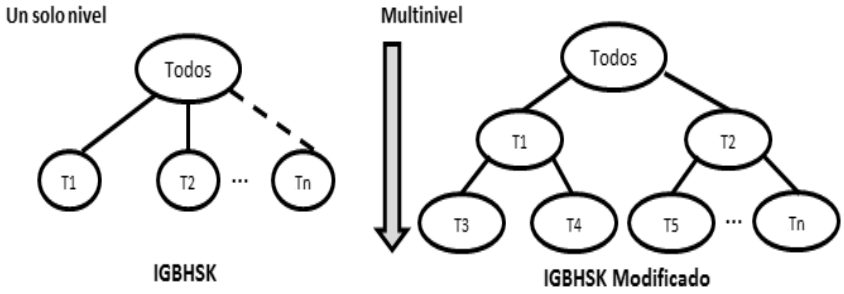


Figura 28. Jerarquía algoritmo IGBHSK

El algoritmo IGBHSK modificado (Anexo I) inicia con el cargue de un conjunto de datos representado como un archivo XML generado por el algoritmo Cosme y su estructura está dada por un nodo principal llamado *resultados*, de este se desprende muchos nodos *Documentos*, cada nodo *documento* contiene a su vez varias nodos que describen atributos necesarios en el proceso de la jerarquía multinivel. *Doc_Url*, el cual representa el Id del documento, *Doc_Titulo* es el nombre del título, *Doc_texto* es el texto original del documento, *Doc_Texto_Revisado* es el texto original después de que se le aplica un proceso de pre procesamiento, *Doc_Sin_Palabras_Vacias* es el texto original pre procesado pero después de aplicarle un pre procesamiento de análisis léxico, eliminación de palabras vacías y steaming, *Doc_Terminos* son todos los términos encontrados en el documento y por ultimo *Doc_Posicion_Palabras_Vacios* representa la posición de las palabras vacías removidas del texto original pre-procesado (ver Figura 29).

```
<Resultados xmlns="http://tempuri.org/Resultados.xsd">
  <Documentos>
    <Doc_Url>1</Doc_Url>
    <Doc_Titulo>A new similarity measure for collaborative filtering to all
    <Doc_Texto>Information Sciences    new similarity measure for collabora
    <Doc_Texto_Revisado> </Doc_Texto_Revisado>
    <Doc_Texto_Sin_Palabras_Vacias> </Doc_Texto_Sin_Palabras_Vacias>
    <Doc_Terminos> </Doc_Terminos>
    <Doc_Posicion_Palabras_Vacios> </Doc_Posicion_Palabras_Vacios>
  </Documentos>
```

Figura 29. Estructura Conjunto de datos

Una vez cargado el respectivo conjunto de datos se valida que el número de documentos en éste sea al menos de cuatro documentos. Se indexa en memoria los documentos, por medio de una librería interna llamada LUCENE.NET [48] en el algoritmo IGBHSK el cual le permite indexar los documentos a través de un índice que se crea en tiempo de ejecución y luego se realiza el pre procesamiento del texto aplicando análisis léxico, eliminación de palabras vacías y lematización por medio de la misma librería [49], este paso solo se realiza para el primer conjunto de documentos, ya que los conjuntos de datos siguientes salen a partir de conjunto inicial de documentos, por consecuencia el texto ya estaría pre procesado, en este mismos paso se realiza la remoción de palabras vacías. Luego se construye la matriz de términos TDM, se lista los términos del documento y se calcula la frecuencia observada de cada término en cada documento (TF o fi) y la máxima frecuencia, para dar paso a la matriz TDF-IDF. Con estos datos se procede a realizar la ejecución del IGBHSK modificado, luego se realiza la generación de

las etiquetas de los grupos para el nivel correspondiente. Por último se realiza un solapamiento de los grupos, determinando en que grupo o grupos de tópicos estará asociado cada documento. Este proceso se repite iterativa mente hasta recorrer todos los conjuntos de datos generados en el nivel anterior. El proceso de solapamiento de los grupos, para el caso de estudio es necesario, ya que los documentos pueden pertenecer a más de un tópico como se modela en la dimensión tópico, razón por la cual estos documentos tienen diferentes probabilidades en cada tópico.

Esta jerarquía se va generando en un árbol de directorios como se muestra en la Figura 30 donde por cada nivel se crea físicamente una carpeta con un nombre predefinido "*nivel_*" seguido del número del nivel. La jerarquía mostrada en la Figura 32 es tomada de un ejemplo de cargue de 200 documentos el cual generó una jerarquía a tres niveles como se muestra, es de aclarar que dependiendo del número de documentos a procesar, los temas asociados a cada documento y los procesos que utiliza el algoritmo, se puede generar *n* niveles y en cada nivel *n* tópicos asociados. Dentro de cada carpeta se encuentran varios archivos: por ejemplo *docs_grupo_0_nivel_1.xml* representa un nuevo conjunto de datos que contienen los documentos pertenecientes al grupo cero del nivel uno los cuales serán procesados en otra iteración, el archivo *etiquetas_0_del_nivel_2.xml* que son los nombres de las etiquetas generadas por IGBHSK modificado para cada grupo del nivel correspondiente, el archivo *matrizTDFIDF_grupo_0_del_nivel_1.xml* en el cual se encuentra como su nombre lo indica la matriz TDF-IDF que necesita el algoritmo PLSA para poder sacar las respectivas probabilidades de los documentos hacia un tópico o grupo. Y por último se encuentra el archivo *Terminos_0_nivel_0.txt* que solo contiene el número de grupo por nivel, este archivo es usado por el mismo IGBHSK modificado para apoyarse en su proceso de jerarquía multinivel. Como resultado del IGBHSK modificado, se obtiene la jerarquía de tópicos con n niveles, la cual se carga en la dimensión tópico.

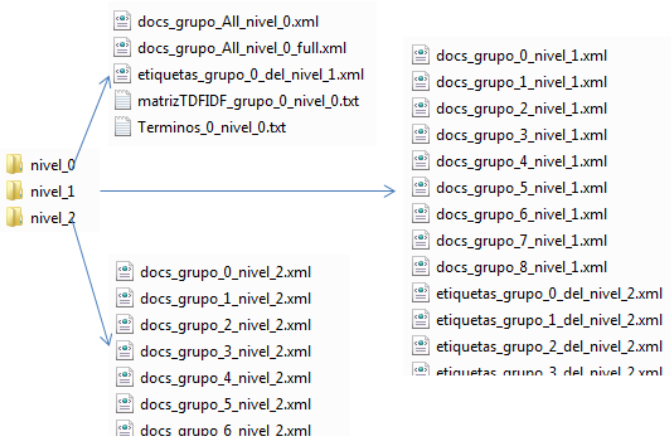


Figura 30. Jerarquía multinivel – para 200 documentos

El algoritmo IGBHSK original fue escogido con base en los estudios y evaluaciones previas realizadas sobre este algoritmo[19], demostrando los buenos resultados en el agrupamiento de los documentos junto con su etiquetado [8], entre las características más importantes, y de acuerdo a las necesidades del proyecto, este algoritmo realiza

[19] Proyecto de investigación *Hibridación de la Mejor Búsqueda Armónica Global y el Algoritmo K-means para el Clustering de Documentos Web*, realizado por estudiantes de la Universidad del Cauca.

agrupamiento, solapamiento y etiquetado de los grupos, los cuales son requisitos para una jerarquía multinivel planteada en el proyecto. Adicionalmente, este algoritmo genera la matriz de términos por documento necesaria para la creación de las medidas probabilísticas definidas en el modelo multidimensional.

4.1.3 Probabilidades con PLSA

Para obtener las probabilidades se utilizó el algoritmo PLSA. De acuerdo al modelo planteado se proponen dos medidas probabilísticas *pTopicDoc* y *pWordTopic* correspondientes a las probabilidades que se obtienen de PLSA, *P(z|d)* y *P(w|z).* Esto permitirá asociarle a cada documento de la jerarquía de tópicos definida, una probabilidad de importancia o relevancia. De igual forma se hace para las palabras o términos que se relacionan a cada tópico de la jerarquía.

Para la ejecución del algoritmo PLSA se utilizó la implementación de HANG CHEN [50] de la Universidad de Illinois, que fue desarrollada en lenguaje C++. Como entrada de datos la implementación de PLSA requiere de la matriz de co-ocurrencia de términos generada por el algoritmo IGBHSK para hallar las probabilidades *P(w|z)* y *P(z|d).* Este proceso se hace para cada tópico del penúltimo nivel de la jerarquía de tópicos definida. Con base en lo anterior, se desarrolló una aplicación que recibe la matriz y el número de tópicos (Anexo G). Por ejemplo, en la Figura 31 se observa el proceso que permite obtener las probabilidades.

Solo para efectos de ejemplo, suponga la jerarquía mostrada en la Figura 31 donde se carga el conjunto de archivos del nivel 1 (o dependiendo del nivel de profundidad de la jerarquía de tópicos, se carga los archivos del penúltimo nivel) correspondiente a las matrices de término por documento de cada tópico, para este ejemplo serán las matrices de los tópicos T2 y T3.

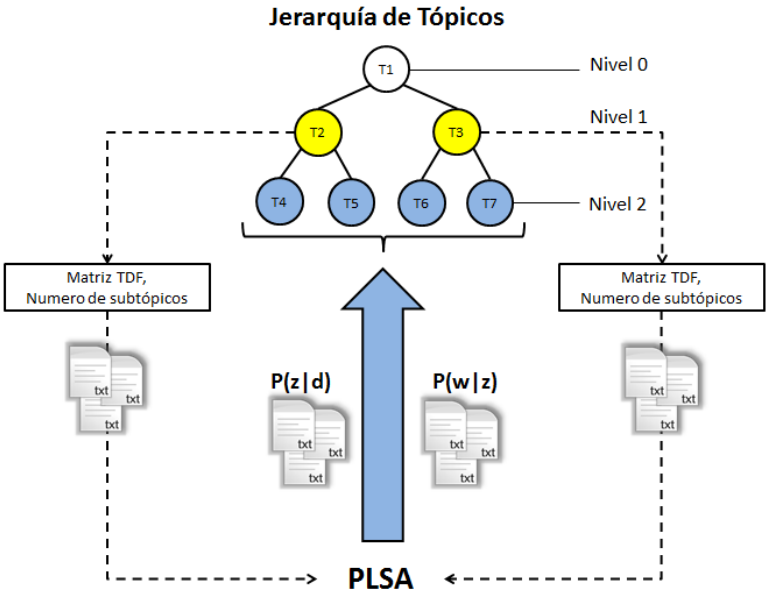


Figura 31. Proceso para obtener las probabilidades P(z|d) y P(w|z)

Seguidamente se carga el conjunto de archivos del nivel 1 (o dependiendo del nivel de profundidad de la jerarquía de tópicos, se carga los archivos del penúltimo nivel) correspondiente a las etiquetas de los sub tópicos del nivel 2 (o las etiquetas del último nivel, si se tiene otro nivel de profundidad). Con esta carga se observa cuántos subtópicos tienen los tópicos del penúltimo nivel, para este ejemplo el tópico T1 tiene dos subtópicos T4 y T5, y el tópico T2 tiene dos sub-tópicos T6 y T7.

Al tener cargados este conjunto de archivos, se ejecuta PLSA a cada tópico del nivel 1 (o dependiendo del nivel de profundidad de la jerarquía de tópicos, se ejecuta PLSA a cada tópico del penúltimo nivel) arrojando como resultado los siguientes archivos.

- Un conjunto de archivos TXT donde se especifica las probabilidades de los tópicos en los documentos correspondientes a cada tópico del último nivel. Es decir un archivo para T4 donde contiene las probabilidades P(z|d) (*pTopicDoc),* de igual forma para T5, T6 y T7.

- Un conjunto de archivos TXT donde se especifica las probabilidades de las palabras en los tópicos del último nivel. Es decir un archivo para T4 donde contiene las probabilidades P(w|z) (*pWordTopic),* de igual forma para T5, T6 y T7.

Este conjunto de archivos se encuentran almacenado en una carpeta con nombre *prob*, el cual será utilizado para el cruce de la información de los documentos requerido en el proceso ETL, es importante recordar que los documentos serán asociados al último nivel de la jerarquía de tópicos.

4.1.4 Proceso ETL

Teniendo ya definido el conjunto de dimensiones y de tablas de hecho de la bodega de documentos, se deben estructurar formalmente los procesos que permitirán poblar la bodega de documentos desde la fuente de datos, en este caso de una base de datos relacional de SQL Server 2008. Se definieron procesos que permitieron mapear los datos desde el sistema fuente correspondiente hacia la bodega de documentos, estos procesos son llamados ETL.

El proceso ETL se inició ordenando las tablas y secuenciando las transformaciones para cada conjunto de datos, todas las tablas de dimensión fueron cargadas antes que las tablas de hecho. El desarrollo de la aplicación ETL se inició con la dimensión más simple y se continuó con las demás hasta llegar a las tablas de hecho. De acuerdo con esto y con los archivos XML generados en los pasos 1 y 2 del algoritmo Cosme (ver Figura 27 y Figura 27), se pobló las dimensiones y las tablas de hecho correspondientes al modelo multidimensional propuesto (Anexo L).

4.1.5 Visualización OLAP

Para el análisis y la visualización de los datos cargado en las tablas del modelo multidimensional, se utilizó la herramienta OLAP *Dundas Chart for ASP.NET*, la cual junto con la tecnología *ASP.Net* de Microsoft, permite generar una interfaz web (Anexo H) para

que el usuario final pueda visualizar el cubo OLAP correspondiente al modelo multidimensional propuesto, incluido la ejecución de las operaciones tradicionales OLAP sobre las medidas textuales definidas en el capítulo 3.2 .

Para mostrar esta visualización OLAP, se seleccionó un conjunto de documentos científicos conformado por 200 documentos. Este conjunto pasa por los procesos definidos en la arquitectura general (capítulo 3.1), donde fueron procesados, clasificados, ponderados y cargados en el modelo multidimensional. La Figura 32 muestra cómo se estructuro la jerarquía de tópicos, donde el nivel 0 al tener un solo tópico se define como todos, el nivel 1 tiene nueve tópicos y para el nivel 2 que es el último nivel de esta jerarquía se generaron treinta y un tópicos, para los 200 documentos utilizados en este cargue de datos. Esta jerarquía es almacenada en la dimensión tópico como una jerarquía organizacional de tres niveles.

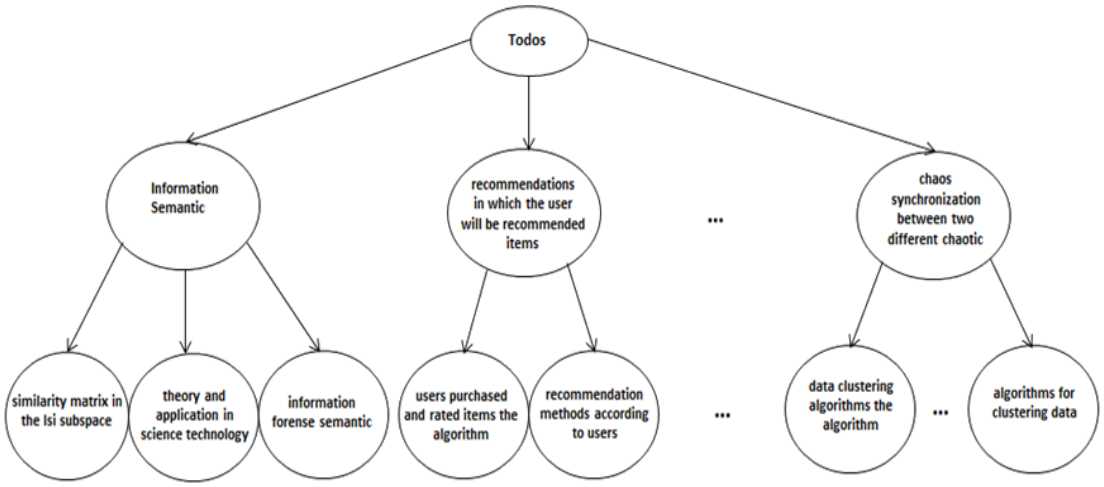


Figura 32. Jerarquía de Tópicos

En la Figura 33 se muestra la visualización de la información de los 200 documentos, donde las medidas textuales con su función de agregación (promedio) trabajan de igual forma a las medidas tradicionales en un cubo OLAP. Como se observa, en el lado izquierdo se encuentran las dimensiones del modelo multidimensional, al lado derecho se tienen las medidas textuales (*Documents*, *Words-Prob* y *Topics-Prob*) y una medida numérica (recuento *FactAssigantion*, que permite ver el número de documentos que se han procesados) que genera automáticamente Analysis Services. En el centro de la figura se visualiza el cruce de la información donde se puede arrastrar dimensiones de acuerdo a una medida marcada. De acuerdo a esto las consultas más interesantes sobre el modelo, son las que implican las medidas textuales, ya que estas permiten la navegación por medio de las jerarquías definidas en las dimensiones implicadas en la consulta.

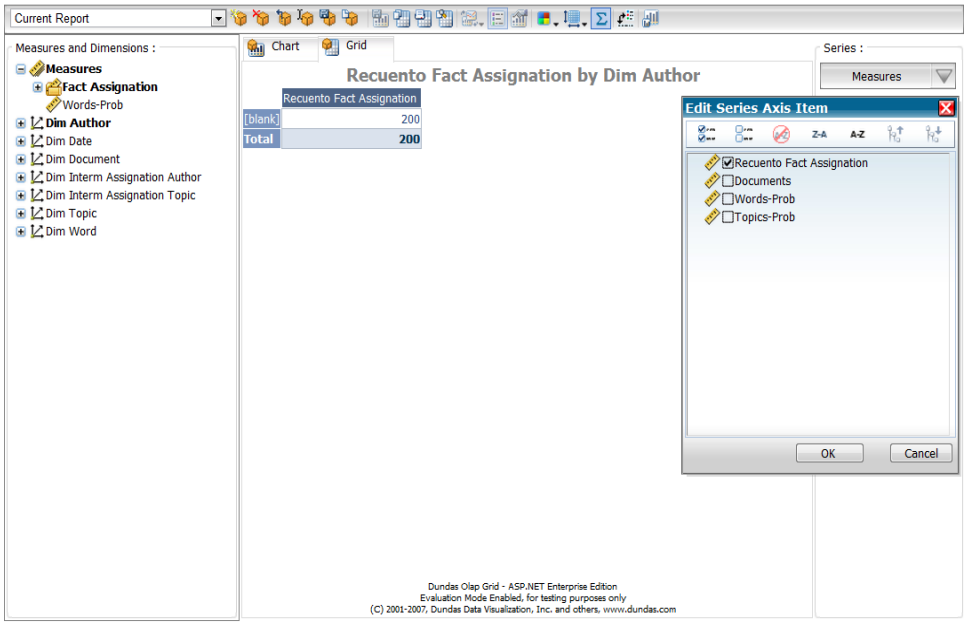


Figura 33. Herramienta Dundas OLAP Grid

Para tener una idea de cómo trabaja una medida numérica sobre un cubo OLAP, se presenta en la Figura 34 la medida *Recuento FactAssignation*con la jerarquía de tópicos, esta medida es de tipo entero y su función de agregación por defecto es la suma. Haciendo un análisis de esta consulta se puede observar cuáles son los tópicos con mayores documentos.

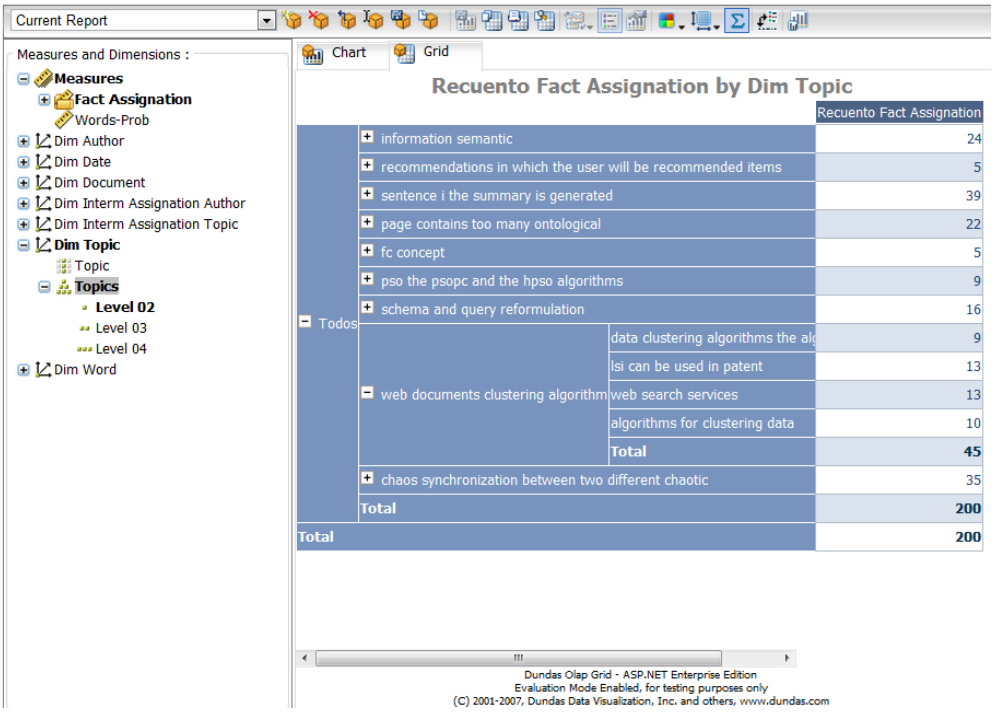


Figura 34. Número de documentos en la jerarquía de tópicos

Ahora en la Figura 35 se muestra cómo funciona la medida textual *Topics-Prob*, al consultar los tópicos con sus probabilidades en la jerarquía definida en la dimensión documento (Tipo de documento → Documento), permitiendo obtener un análisis de cuales tópicos son más relevantes, de acuerdo a su porcentaje, en los diferentes tipos de documentos o en el documento como tal.

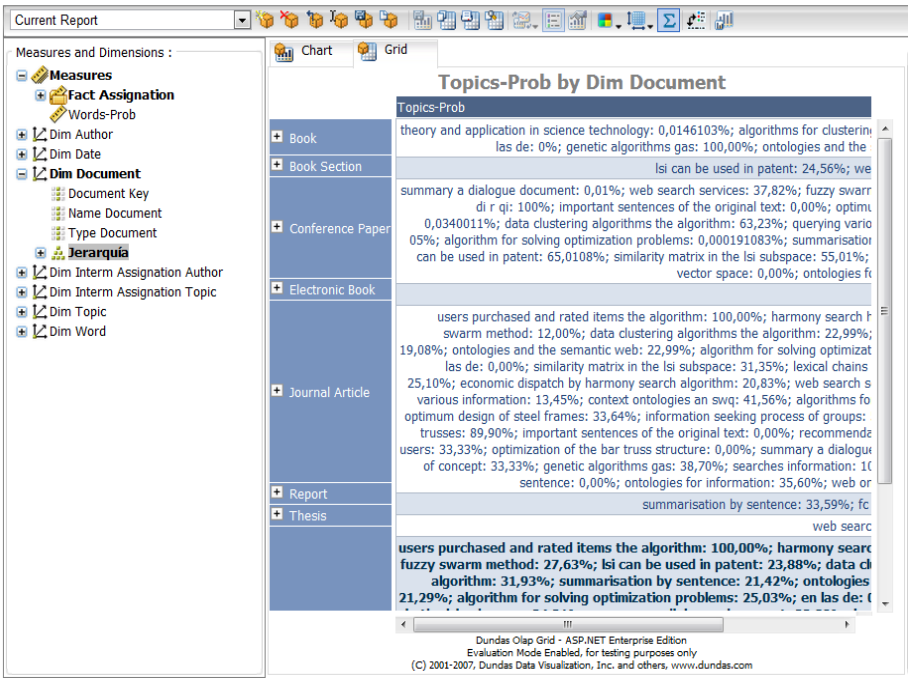


Figura 35. Tópicos con sus probabilidades en la jerarquía de la dimensión documento.

En la Figura 36 se muestra una consulta que permite obtener los tópicos con sus probabilidades de acuerdo a la jerarquía definida en la dimensión fecha, obteniendo los tópicos por año-mes-día. Haciendo un análisis de la consulta se pueden ver los tópicos o temas más tratados, de acuerdo a su porcentaje, con respecto a un año, mes o día en particular. De igual forma se pueden realizar otras consultas que impliquen el uso de la medida textual en la tabla de hecho principal.

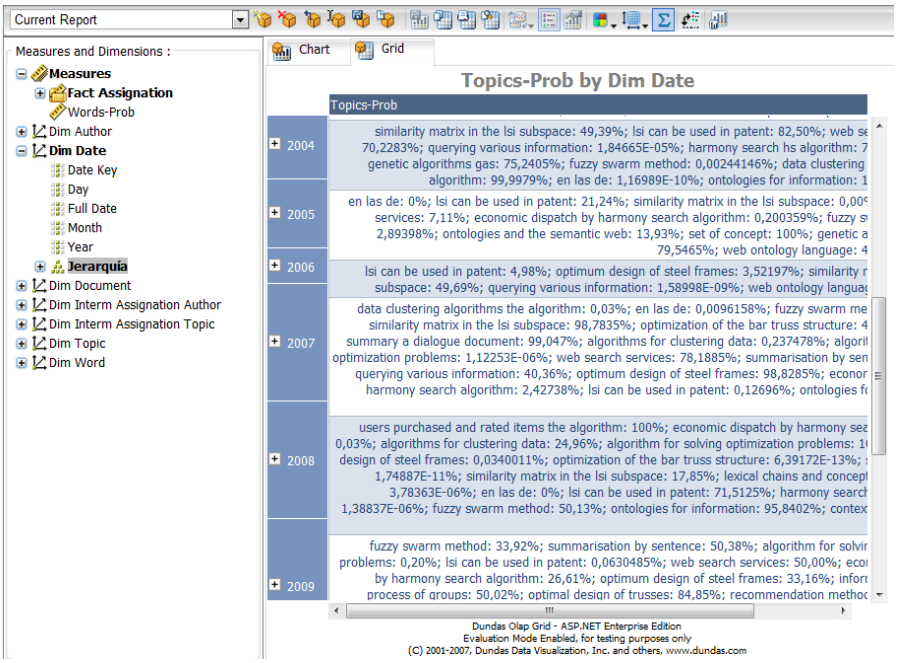


Figura 36. Tópicos con sus probabilidades en la jerarquía de la dimensión fecha.

Por otra parte la medida textual *Documents* muestra los documentos con sus probabilidades de acuerdo a los tópicos definidos en la dimensión tópicos. En la Figura 37 se puede observar como en la jerarquía de tópicos se tiene un conjunto de documentos asociados a cada tópico, permitiendo analizar cuáles son los documentos más relevantes en un tópico de acuerdo a su porcentaje.

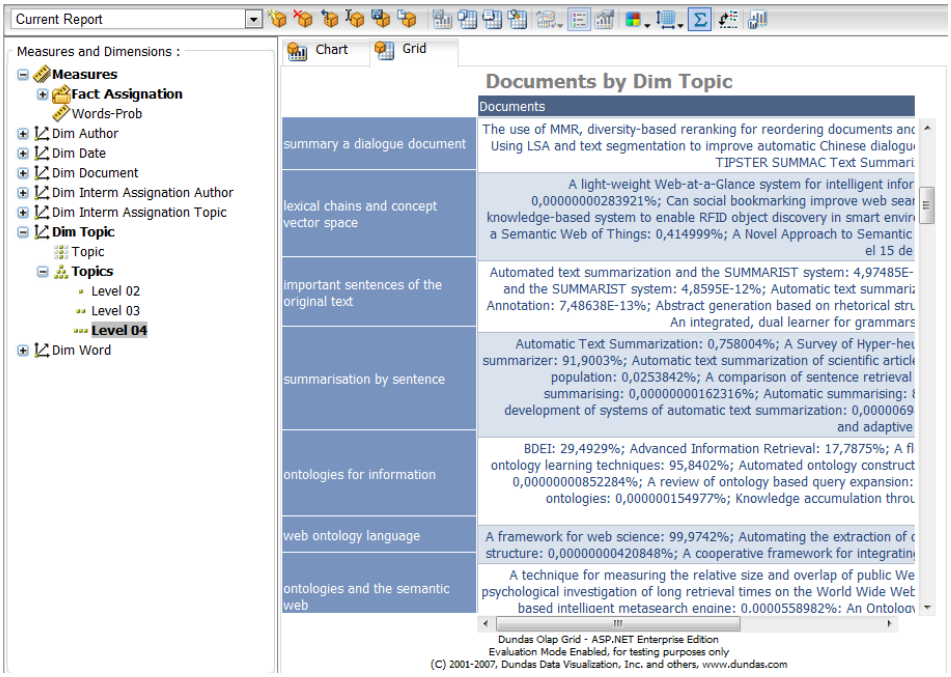


Figura 37. Documentos con sus probabilidades en la jerarquía de tópicos.

4.2 RESULTADOS DE LA EVALUACIÓN

La evaluación del modelo se realizó en dos partes, la primera fue con respecto al tiempo de ejecución de cuatro tipos de consultas y la segunda se llevó a cabo por medio de un análisis estadístico de los resultados de una encuesta aplicada a usuarios finales.

4.2.1 Evaluación del tiempo de consulta sobre el modelo multidimensional

Para esta primera etapa de evaluación se tomó el conjunto de documentos de 729 artículos científicos en formato pdf de una librería llamada EndNote[20] usada por algunos profesores del departamento de sistemas de la Universidad del Cauca. Para la prueba los documentos se cargaron en el modelo en grupos de 200, 400 y 600 documentos, para cada grupo de prueba se tomó el tiempo de respuesta de cuatro tipos de consultas en el modelo multidimensional. Para la selección de estas consultas se tuvo en cuenta la navegación por las jerarquías de dimensión y la navegación por una y dos dimensiones.

La herramienta OLAP de Dundas, se adaptó y configuró para tomar el tiempo de respuesta de las consultas. Las características del computador en el cual se realizaron las pruebas y se tomaron los tiempos fueron: un procesador de cuatro núcleos a 2.8 Gz, y cuatro gigas de Ram; además se usó el motor de base de datos Sql Server 2008 y Bussines Intelligence Development Studio (Analysis Services e Integrations Services 2008).

Cada una de estas consultas se realizó sobre los tres grupos de documentos escogidos de la librería EndNote, la captura de los tiempos se hizo en milisegundos y usando la herramienta incorporada en SqlServer 2008 Development llamada SQl Server profiler 2008 obteniendo los siguientes resultados:

- ✓ **Consultas**

Se evaluó el tiempo de respuesta del modelo propuesto con dimensiones que presentan jerarquías como: *DimDate* (con tres niveles), *DimDocument* (con dos niveles) y *DimTopic* (con tres niveles); para el análisis de los resultados obtenidos se elaboraron gráficas en las que se relacionan, el tiempo de respuesta (obtenidos en la herramienta OLAP y medidas en milisegundos) respecto al número de documentos procesados en cada nivel de las jerarquías. Este proceso cuenta con dos tipos de consulta sobre la bodega, que son:

- Una dimensión:
  - Consulta 1: La jerarquía de la Dimensión *Document* junto con la medida de texto *TextMeasure_Topics_Probab*.
  - Consulta 2: La jerarquía de la Dimensión *Date* junto con la medida de texto *TextMeasure_Topics_Probab*.
  - Consulta 3: La jerarquía de la Dimensión *Topic* junto con la medida de texto *TextMeasure_Documents*.

[20]un gestor de referencias que ayuda a los investigadores en el manejo de sus bibliografías

- Dos Dimensiones:
  - Consulta 4: La combinación de las jerarquías de la Dimensión *Document* y *Date* junto con la medida de texto *TextMeasure_Topics_probab*.

✓ Gráficas con una Dimensión

En la Figura 38 se muestra el tiempo de ejecución (en milisegundos) para la jerarquía de la dimensión *DimDocument* y la medida textual *TextMeasure_Topics_Probab*. Se observa un comportamiento lineal en el mismo nivel de la jerarquía cuando aumenta el número de documentos.

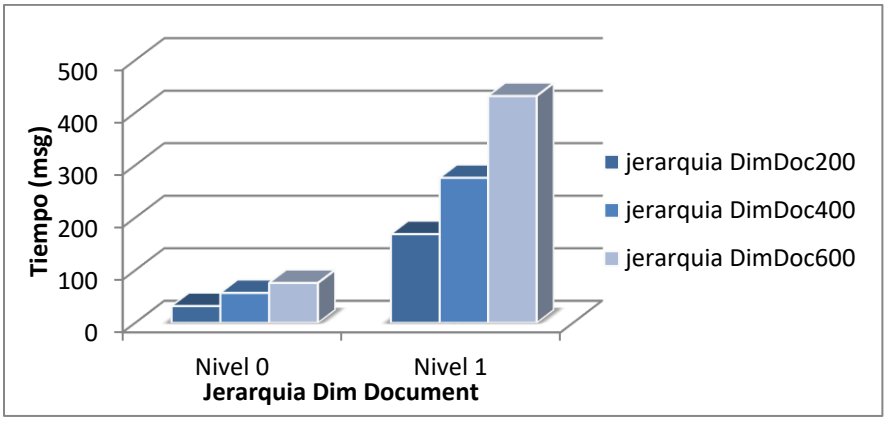


Figura 38. Jerarquía DimDocument vs Tiempo (milisegundos)

En la Figura 39 la consulta se hace sobre la jerarquía de la dimensión *DimDate* y la medida textual *TextMeasure_Topics_Probab*. Se observa un comportamiento lineal entre cada nivel de la jerarquía cuando aumenta el número de documentos.

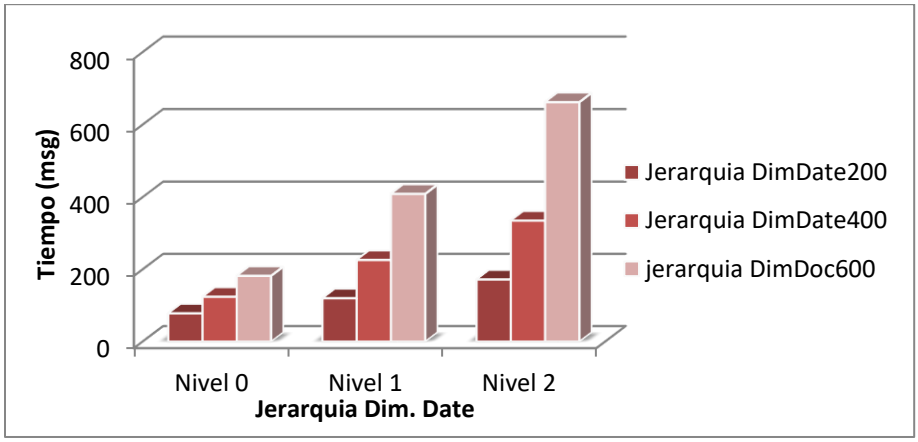


Figura 39. Jerarquía DimDate vs Tiempo (milisegundos)

Y en la Figura 40, la consulta se hace sobre la jerarquía de la dimensión *DimTopic* y la medida textual *TextMeasure_Documents*. También se observa un comportamiento lineal entre cada nivel de la jerarquía cuando aumenta el número de documentos.

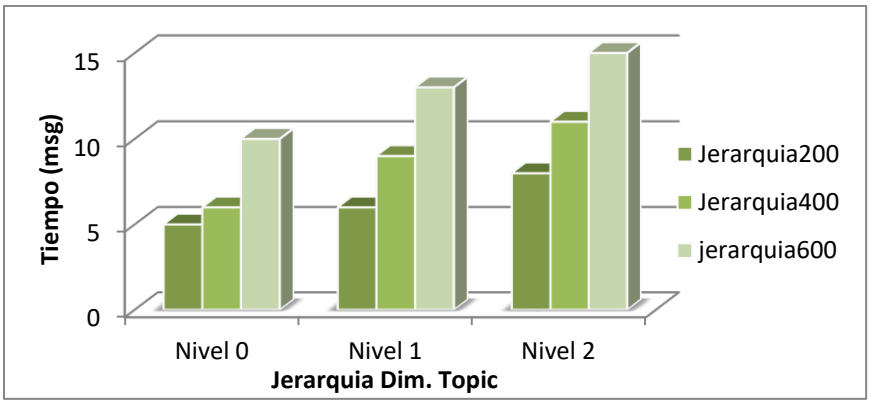


Figura 40. Jerarquía DimTopic vs Tiempo (milisegundos)

✓ **Gráficas con dos Dimensiones**

En el caso de las consultas con dos dimensiones, una consulta se puede realizar de dos formas: la primera, acoplada, es aquella donde las jerarquías de las dimensiones son arrastradas sobre un mismo eje de la grilla de la herramienta OLAP; la segunda, por medio de una matriz, es aquella donde las jerarquías de las dimensiones son arrastradas sobre diferentes ejes de la grilla de la herramienta OLAP.

En las gráficas de la Figura 41 y la Figura 42 se puede observar que es mucho más rápido el tiempo de consulta por parte del usuario si se realiza una búsqueda con el esquema acoplado. En este esquema se tiene un tiempo aproximado de 2500 milisegundos para 600 documentos, entregando los resultados al usuario en un menor tiempo, ya que como se muestra en la consulta por matriz los tiempos de respuesta para 600 documentos están alrededor de 100.000 milisegundos. La consulta por matriz tarda más tiempo debido a que el cruce de la información también se realiza sobre los campos nulos o vacíos.

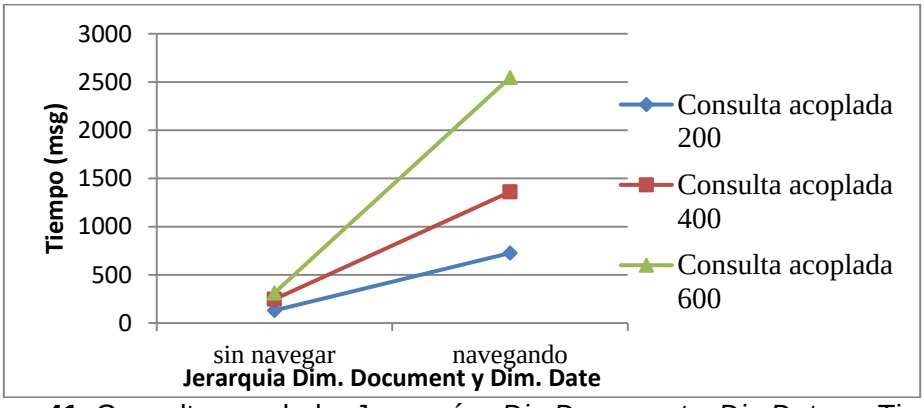


Figura 41. Consulta acoplada: Jerarquías *DimDocument* y *DimDate* vs Tiempo (milisegundos)

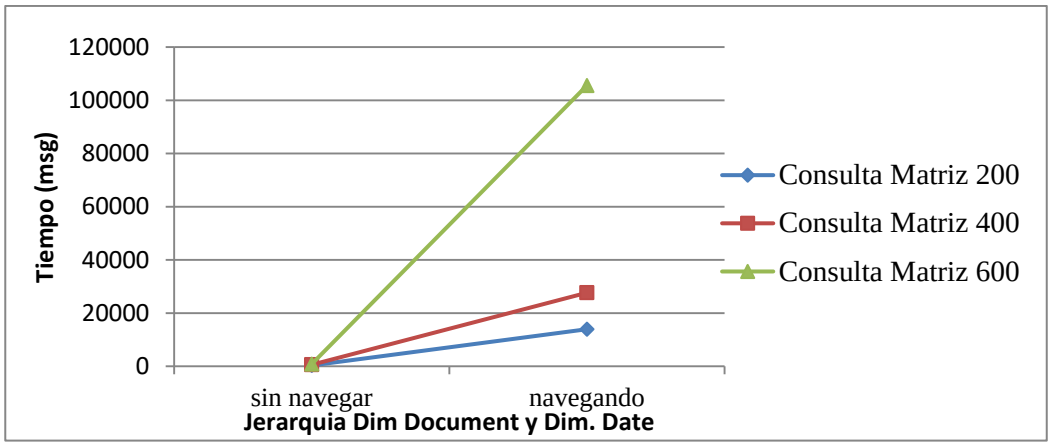


Figura 42. Consulta matriz: Jerarquías *DimDocument* y *DimDate* vs Tiempo (milisegundos)

✓ **Proyección**

En las gráficas anteriores se evidencia un comportamiento lineal al presentarse incrementos del tiempo a medida que se aumentaron el número de documentos a procesar. Considerando este comportamiento y con el objeto de elaborar una proyección aproximada del tiempo de respuesta esperado al procesar 1000 documentos, se aplicó sobre cada grupo de datos el método de mínimos cuadrados, el cual permitió obtener la ecuación que relaciona las dos variables evaluadas, a continuación se presentan las proyecciones para las diferentes consultas inicialmente ejecutadas. La aplicación de estas ecuaciones para los 1000 documentos permite visualizar en las primeras graficas (ver Figura 43, Figura 44 y Figura 45) que de acuerdo al nivel de profundidad de la jerarquía el incremento en los tiempos son mínimos.

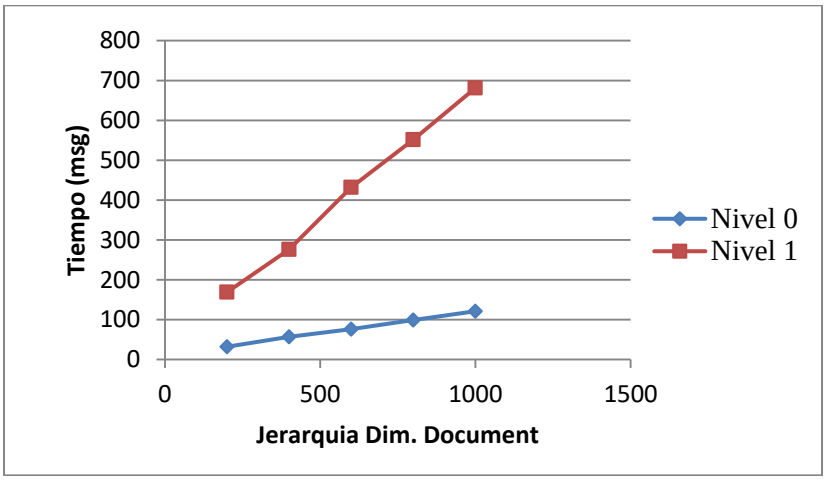


Figura 43. Proyección en 1000 documentos jerarquía *DimDocument* vs Tiempo (milisegundos)

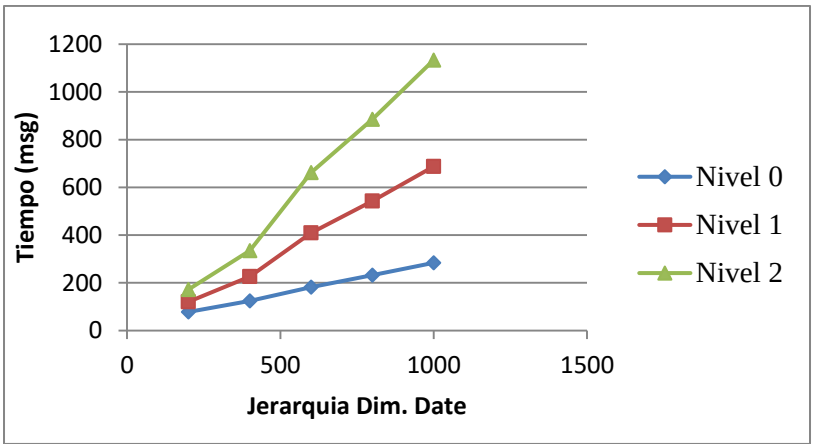


Figura 44. Proyección en 1000 documentos jerarquía *DimDate* vs Tiempo (milisegundos)

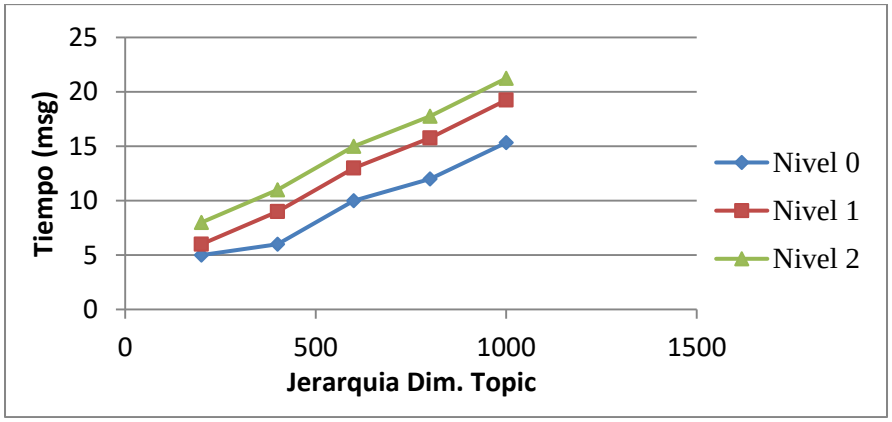


Figura 45. Proyección en 1000 documentos jerarquía *DimTopic* vs Tiempo (milisegundos)

En las gráficas de la Figura 46 y Figura 47 se muestra que de acuerdo al nivel de profundidad de las jerarquías el incremento en los tiempos aumenta pero razonablemente en el tiempo de respuesta de una consulta del usuario final.

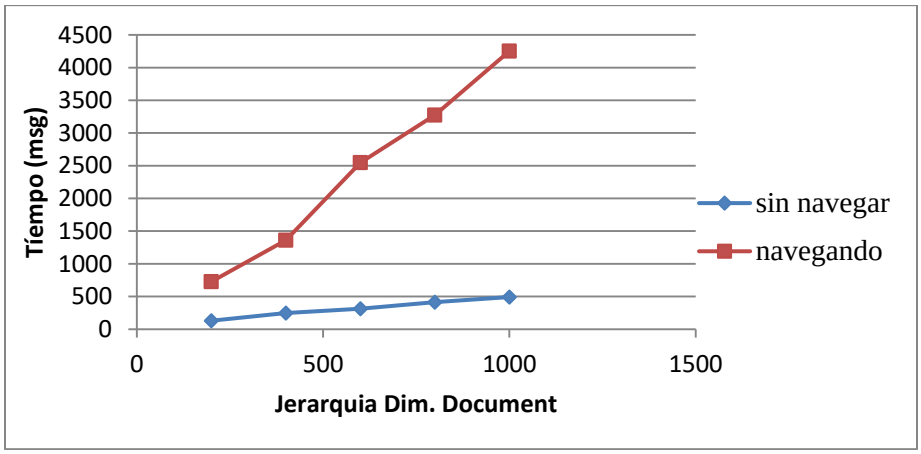


Figura 46. Proyección en 1000 documentos consulta acoplada: Jerarquías *DimDocument* y *DimDate* vs Tiempo (milisegundos)

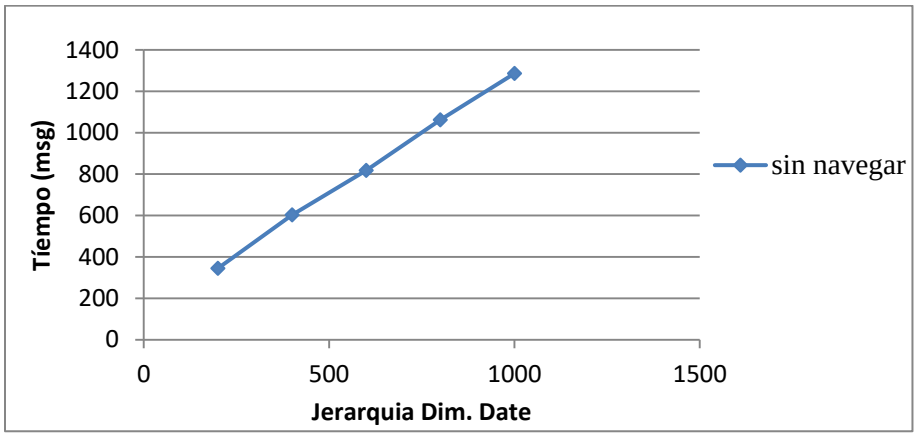


Figura 47. Proyección en 1000 documentos consulta matriz: Jerarquías *DimDocument* y *DimDate* vs Tiempo (milisegundos) sin navegar.

Por su parte en la gráfica de la Figura 48 se puede observar un incremento mucho mayor ya que este tipo de consulta, como antes se había mencionado, retorna el cruce de la información también para campos nulos o vacíos.

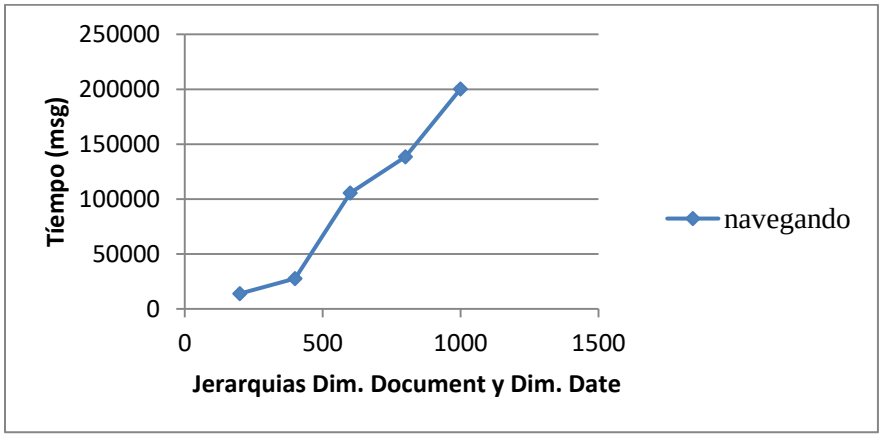


Figura 48. Proyección en 1000 documentos consulta matriz: Jerarquías *DimDocument* y *DimDate* vs Tiempo (milisegundos) navegando.

Es importante mencionar que los valores encontrados mediante este método son una aproximación, considerando que el rango evaluado se encuentra muy distante respecto al valor extrapolado, y la ecuación hallada proviene de una cantidad mínima de datos, no es posible garantizar que el modelo propuesto siga el mismo comportamiento para 600 documentos que para 1000 documentos o más. Es necesario que el modelo se evalué con más documentos para contar con una mejor precisión de los resultados proyectados.

4.2.2 Evaluación de la satisfacción del usuario final

Para esta segunda evaluación se realizó una encuesta tipo test (Anexo K), para evaluar la satisfacción del usuario final en dos factores: la facilidad de uso y tiempos de consulta; basados en el modelo de Torkzadeh and Doll (1999) [22]. Las preguntas fueron

agrupadas de la siguiente manera: en cinco de ellas evalúan la facilidad de uso, tres evalúan el tiempo de consulta y una donde se evalúa en términos generales el modelo planteado. Se evaluó según la escala presentada a continuación:

[1]. Totalmente en desacuerdo (TD)
[2]. En desacuerdo (ED)
[3]. Ni acuerdo ni en desacuerdo (ND)
[4]. De acuerdo (DA)
[5]. Totalmente de acuerdo (TA)

El grupo de encuestados fueron doce estudiantes que se encuentran realizando trabajo de grado en el programa de Ingeniería de Sistemas de la Universidad del Cauca, ellos usaron la herramienta OLAP para cada grupo de prueba (200, 400 y 600 documentos). Esta evaluación se divide en dos partes: la primera, es la realización de un análisis de frecuencias estadísticas de la satisfacción del usuario final en los dos factores y la segunda, un análisis factorial.

4.2.2.1 Análisis de frecuencias estadísticas

Los resultados obtenidos con la primera evaluación se presentan a continuación:

Factor Facilidad de Uso

Para la pregunta 1, *"¿El sistema presenta una fácil navegación?"*, se observa que la tendencia de los encuestados en las cuatro consultas es estar de acuerdo o totalmente de acuerdo con que el sistema presenta una fácil navegación por las jerarquías de las dimensiones de cada consulta al obtener los mayores porcentajes, sin embargo el 8% de los encuestados en cada una de las consultas C3 y C4 se encuentra en una posición de indiferencia ante la fácil navegación, finalmente el 8% de los encuestados en la consulta C2 están en desacuerdo, ver Figura 49. Esto es debido a que cuando se navega por la herramienta OLAP sobre la jerarquía de fecha, no todos los documentos tienen los meta datos para ser clasificados dentro de estas jerarquías, razón por la cual la herramienta OLAP los ubica en un atributo interno llamado *blank*, confundiendo un poco la navegación de los encuestados.

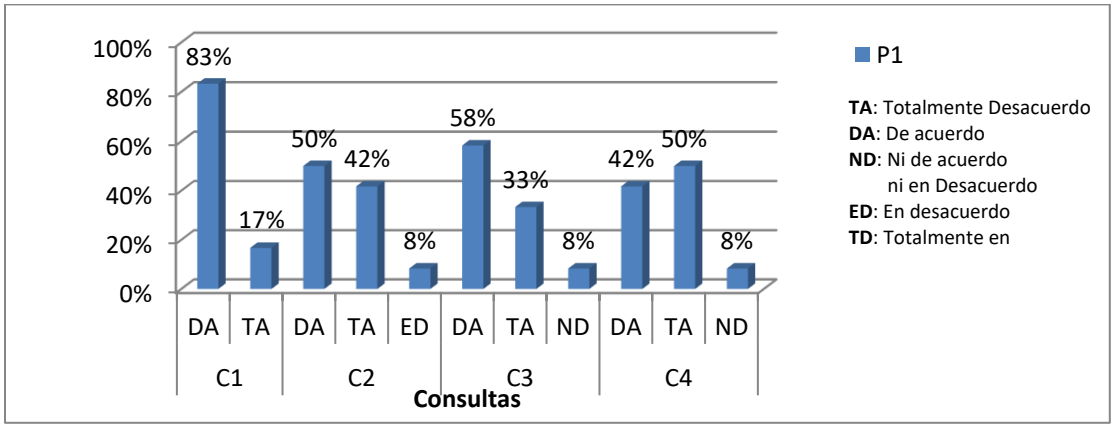


Figura 49. Resultados para la pregunta 1 en las cuatro consultas (Porcentajes)

Para la pregunta 2 "¿Es entendible la presentación de la medida textual para el análisis?", se observa que la tendencia de los encuestados en las cuatro consultas es estar de acuerdo con que es entendible la presentación de la medida textual en cada consulta al obtener los mayores porcentajes, sin embargo, entre el 8% y 25% de los encuestados en cada una de las consultas se encuentra en una posición de indiferencia ante la presentación de la medida textual, finalmente entre el 8% y el 17% de los encuestados en las consultas C2 y C3 están en desacuerdo, ver Figura 50. Esto debido a que algunos de los datos o palabras mostradas en la herramienta no reflejan ningún significado. Esta anomalía se debe a que los documentos cargados contenían algunos caracteres extraños, frases o conjunto de palabras unidas (sin espacios) generado en el proceso de transformación de pdf a txt. Aunque estos caracteres extraños en su mayoría fueron removidos por algoritmo Cosme Paso1.

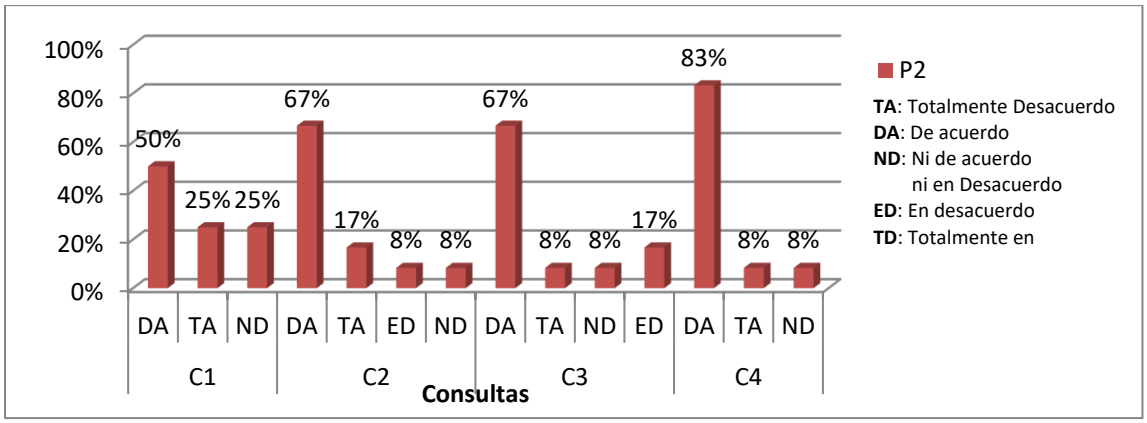


Figura 50. Resultados para la pregunta 2 en las cuatro consultas (Porcentajes)

Para la pregunta 3 "¿El formato de presentación de la medida textual es adecuado?", se observa que la tendencia de los encuestados en las cuatro consultas es estar de acuerdo con que la presentación de la medida textual es el adecuado en cada consulta al obtener los mayores porcentajes, sin embargo, entre el 8% y 25% de los encuestados en cada una de las consultas se encuentra en una posición de indiferencia ante sí la presentación de la medida textual es el adecuado, finalmente 17% de los encuestados en las consultas están en desacuerdo, ver Figura 51.

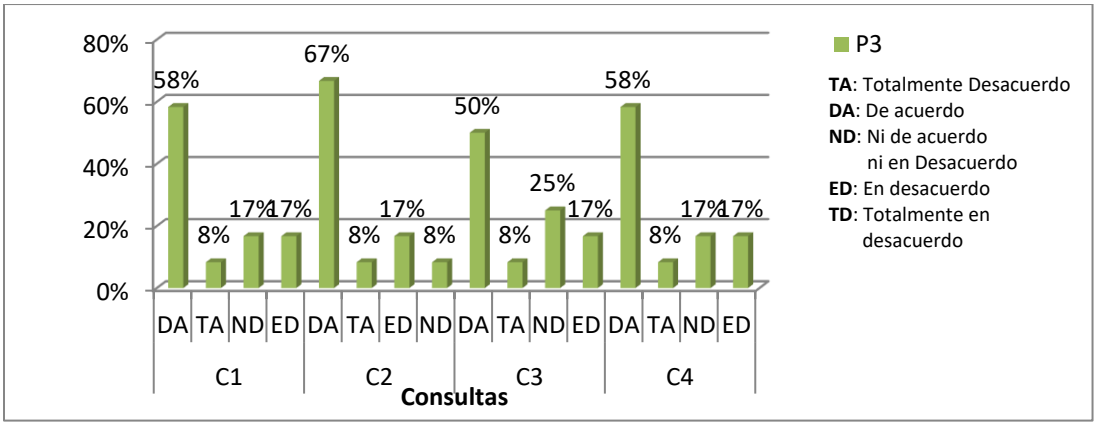


Figura 51. Resultados para la pregunta 3 en las cuatro consultas (Porcentajes)

Para la pregunta 4 "¿La información presentada por la medida textual satisface sus expectativas?", se observa que la tendencia de los encuestados en las cuatro consultas es estar de acuerdo con las expectativas de la medida textual en cada consulta al obtener los mayores porcentajes, sin embargo, el 8% de los encuestados en las consultas C1, C2 y C3 se encuentra en una posición de indiferencia ante expectativa de la medida textual, finalmente entre el 17% y 25% de los encuestados en las consultas están en desacuerdo, ver Figura 52.

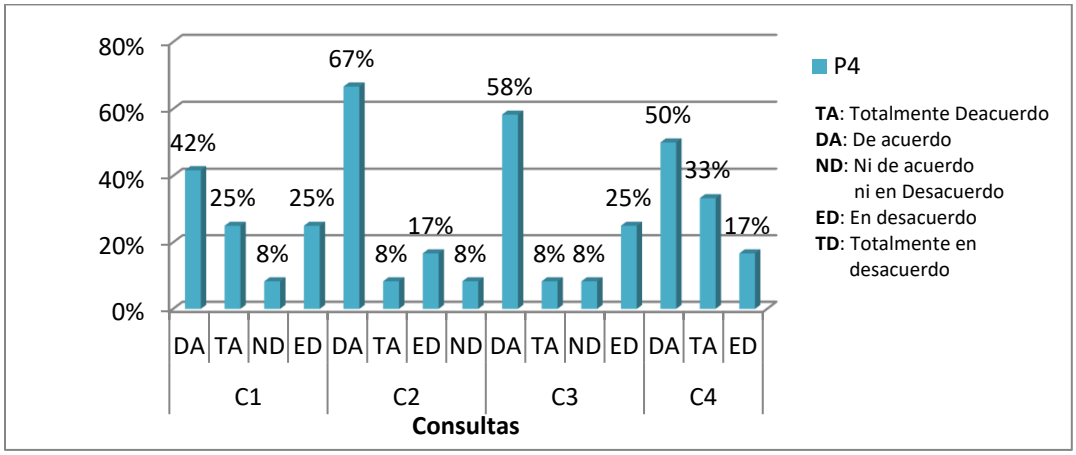


Figura 52. Resultados para la pregunta 4 en las cuatro consultas (Porcentajes)

Para la pregunta 5 "¿Es entendible la relación entre la medida textual y la(s) jerarquía(s) de la(s) dimensión(es)?", se observa que la tendencia de los encuestados en las cuatro consultas es estar de acuerdo con las expectativas de la medida textual en cada consulta al obtener los mayores porcentajes, sin embargo, entre el 17% y 25% de los encuestados en las consultas C3 y C4 se encuentra en una posición de indiferencia ante expectativa de la medida textual, finalmente entre el 8% y 17% de los encuestados en las consultas C1, C2 y C3 están en desacuerdo, ver Figura 53.

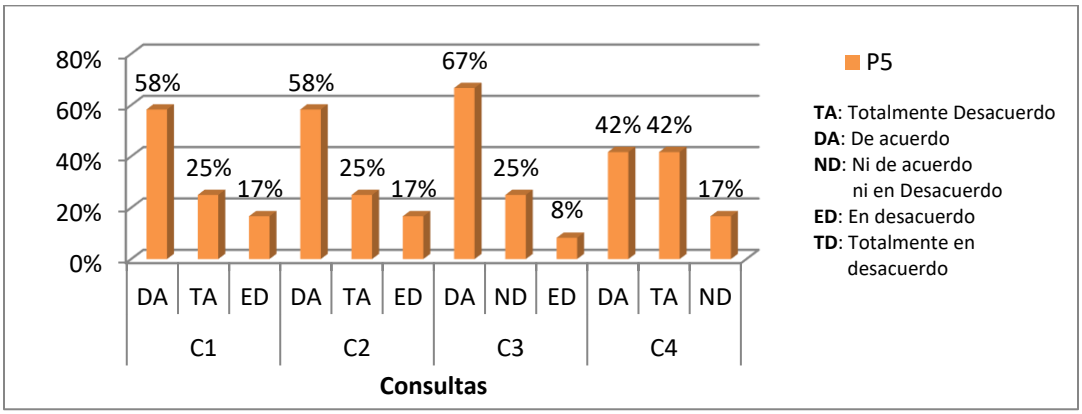


Figura 53. Resultados para la pregunta 5 en las cuatro consultas (Porcentajes)

La razón del comportamiento de los encuestados en desacuerdo y los que no asumen una posición en las preguntas 3, 4 y 5 es debido a que las herramientas OLAP actuales

no están diseñadas para mostrar medidas textuales lo que imposibilita una adecuada lectura de este tipo de medidas.

En términos generales, la tendencia presentada para este factor es buena debido a que 59% de los encuestados están de acuerdo y un 20% está totalmente de acuerdo con la presentación, formato, navegación y análisis de la medida textual creada. Es de resaltar que uno de los grandes inconvenientes actuales para este tipo de medidas en las bodegas de datos son las herramientas OLAP, ya que fueron pensadas inicialmente para el análisis de medidas numéricas dentro de las organizaciones. Por otra parte la calidad de los datos, al igual que en las bodegas de datos normales es una tarea compleja y dispendiosa, por lo que se hace necesario utilizar otros algoritmos de limpieza y depuración, que permitan asegurar más la integridad y calidad de los datos presentados al usuario final.

Factor tiempo de consulta

Para la pregunta 6 "¿El tiempo de respuesta fue lo suficientemente rápido?", se observa que la tendencia de los encuestados en las cuatro consultas es estar totalmente de acuerdo con que el tiempo de respuesta fue rápido en cada consulta al obtener los mayores porcentajes, sin embargo, entre el 8% y 17% de los encuestados en las consultas C3 y C4 se encuentra en una posición de indiferencia ante el tiempo de respuesta, finalmente el 8% de los encuestados en la consulta C4 están en desacuerdo, ver Figura 54.

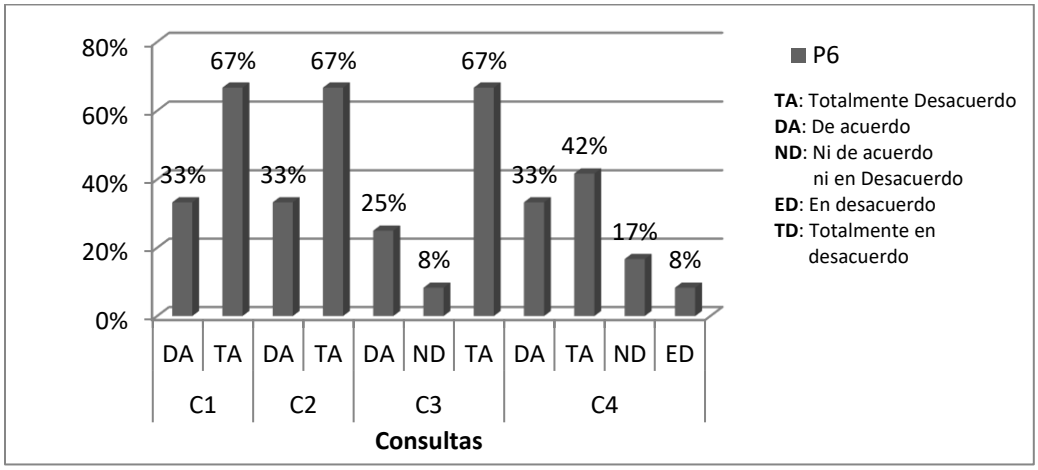


Figura 54. Resultados para la pregunta 6 en las cuatro consultas (Porcentajes)

Para la pregunta 7 "¿El tiempo de respuesta fue el esperado?", se observa que la tendencia de los encuestados en las cuatro consultas es estar totalmente de acuerdo con que el tiempo de respuesta fue el esperado en cada consulta al obtener los mayores porcentajes, sin embargo, entre el 8% y 25% de los encuestados en las consultas C3 y C4 se encuentra en una posición de indiferencia ante el tiempo de respuesta, finalmente el 8% de los encuestados en la consulta C4 están en desacuerdo, ver Figura 55.

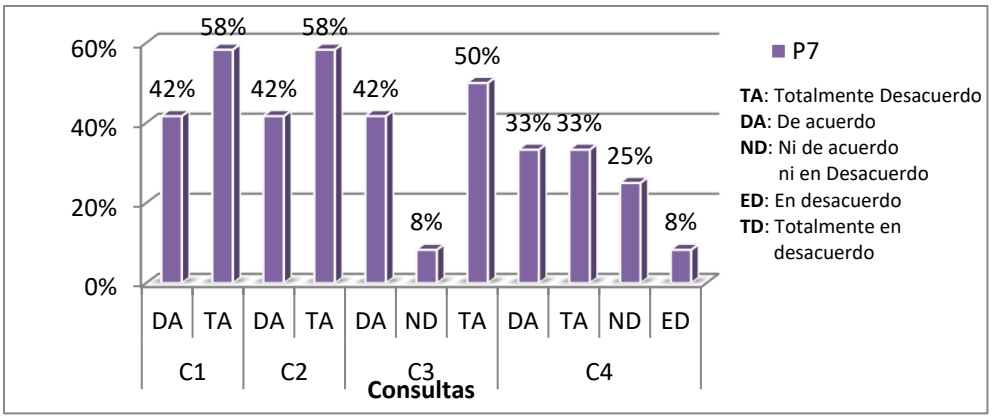


Figura 55. Resultados para la pregunta 7 en las cuatro consultas (Porcentajes)

Para la pregunta 8 "¿El tiempo de respuesta de las consultas se ajusta a sus expectativas?", se observa que la tendencia de los encuestados en las tres primeras consultas es estar totalmente de acuerdo con que el tiempo de respuesta se ajusta a las expectativas, en cada consulta al obtener los mayores porcentajes, sin embargo, entre el 8% y 17% de los encuestados en las consultas C1, C3 y C4 se encuentra en una posición de indiferencia ante el tiempo de respuesta, finalmente el 8% de los encuestados en la consulta C4 están en desacuerdo, ver Figura 56.

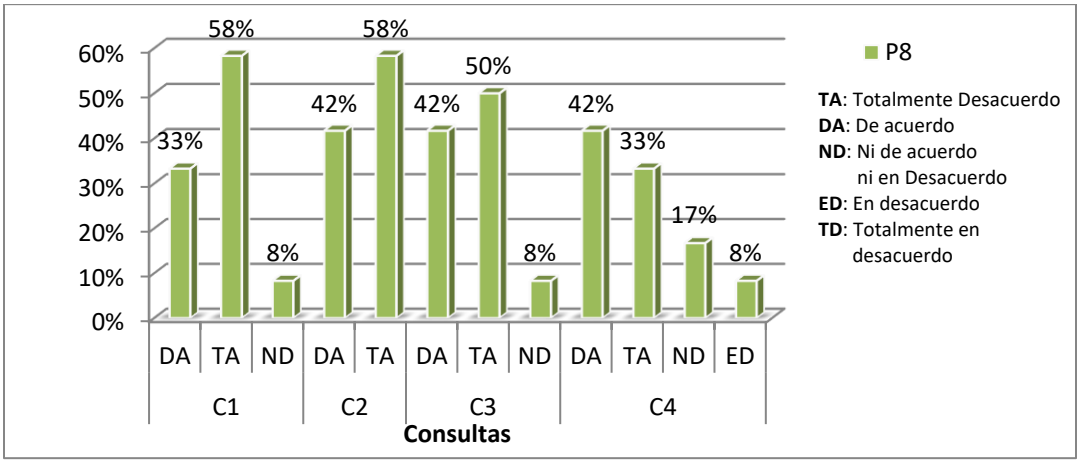


Figura 56. Resultados para la pregunta 8 en las cuatro consultas (Porcentajes)

El comportamiento de los encuestados que no están de acuerdo o no tienen una posición fija en la consulta 4 de las preguntas 6, 7 y 8 se debe a que en esta consulta se involucra dos jerarquías (jerarquía de Documento y Fecha) al mismo tiempo de forma acoplada, esto conlleva a que el tiempo de respuesta sea un poco más lento dependiendo del volumen de datos a procesar.

Es de resaltar que este procesamiento de datos en todas las consultas se realiza una sola vez para todas las jerarquías involucradas en cada una de las consultas, por tal razón la navegación por las jerarquías después de cruzarse la información es muy rápida puesto que la información ya ha sido procesada por primera vez al arrastrar las jerarquías a la herramienta OLAP.

En términos generales este factor, tiene una tendencia bastante buena debido a que el 53% de los encuestados están de totalmente de acuerdo y un 20% está totalmente de acuerdo con el tiempo de respuesta de las medidas textuales. Aunque este factor se analizó con más detalle en la etapa uno del proceso de evaluación (ver capítulo 4.2.1).

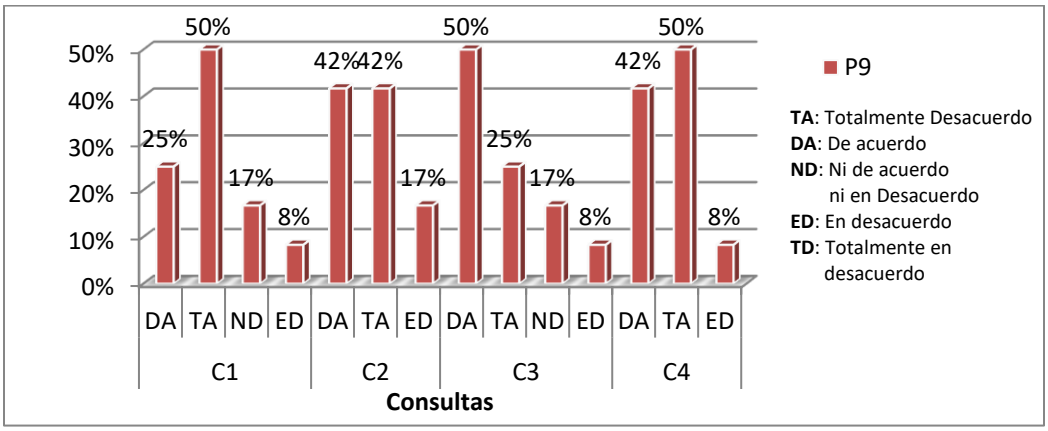


Figura 57. Resultados para la pregunta 9 en las cuatro consultas (Porcentajes)

Por último en esta etapa de evaluación se realizó una pregunta de forma general para conocer la percepción del usuario final en la utilidad en caso de una revisión bibliográfica. La pregunta planteada fue: ¿En su percepción, la navegación en el modelo de documentos a través de las medidas textuales es útil en caso de una revisión bibliográfica? Los resultados obtenidos se muestran en la Figura 57 donde se observa que entre el 25% y 50% de los encuestados están de acuerdo de la utilidad de las medidas textuales en caso de una revisión bibliográfica, sin embargo el 17% de los encuestados en las consultas C1 y C3 no asumen una posición, finalmente, entre el 8% y 17% están en desacuerdo de la utilidad. El comportamiento en desacuerdo es explicado por la percepción del usuario de que la herramienta OLAP no tiene soporte para este tipo de medidas textuales.

Con base en los resultados obtenidos de la encuesta, se hizo un resumen (ver Tabla 2) que permite, en términos generales y teniendo en cuenta la escala de valoración de la encuesta, observar la aceptación al incluir e interactuar medidas textuales con su función de agregación en la herramienta OLAP.

	Facilidad de uso	Tiempo de consulta	Percepción del Usuario final
Totalmente de acuerdo (TA)	21%	**53%**	**42%**
De acuerdo (DA)	**59%**	37%	40%
NI acuerdo ni en desacuerdo (ND)	13%	9%	8%
En desacuerdo (ED)	17%	2%	10%

Tabla 2. Tabla comparativa de porcentajes encuestados

En la tabla anterior se puede observar que un 59% de los encuestados están de acuerdo con la inclusión de medidas textuales con su función de agregación promedio a nivel de facilidad de uso en la navegación a través de las jerarquías. El 17% que están en desacuerdo es debido a varios factores: primero, no todos los documentos cuenta con la información para ser clasificados dentro de una jerarquía. Por ejemplo, cuando se desea consultar los documentos por la jerarquía fecha, algunos documentos no tenían el mes y/o el día, por tanto la herramienta OLAP los muestra en un atributo genérico poco entendible al usuario final. Segundo, la medida textual no se muestra de forma ordenada o con un formato adecuado que permita visualizar la información en forma de lista, esto es debido a que la herramienta OLAP aun no soporta este tipo de medidas textuales dificultando el análisis de los datos. Tercero, algunos términos o frases no son coherentes, o sin sentido, esto debido a que la conversión de los documentos, de PDF a TXT, genera frases o términos unidos y/o caracteres extraños. En general, la tendencia presentada para este factor es buena debido a que los encuestados están de acuerdo con la presentación, formato, navegación y análisis de la medida textual creados.

Para el factor tiempo de consulta un 53% de los encuestados están de acuerdo con el tiempo con que se ejecutan y procesan las. El 9% y 2% de los encuestados que están en una posición neutral y en desacuerdo es debido a varios situaciones: primero, cuando las consultas son acopladas, o que implican más de una dimensión, conllevan a que el tiempo de respuesta sea más lento dependiendo del volumen de datos a procesar. Segundo, cuando las consultas por matriz se ejecutan, el número de celdas que retorna es mayor ya que esta consulta evalúa los campos que son nulos o vacíos. En general, la tendencia es buena teniendo en cuenta que este tipo de herramientas procesan mayor información y más aún cuando se incluye texto.

Para la percepción del usuario final, se encuesto si las medidas textuales podrían servir para una revisión bibliográfica, el 42% de los encuestados están de acuerdo con la inclusión y utilidad de las medidas textuales en el caso de una revisión bibliográfica. El 8% y 10% que están en desacuerdo es debido a que la herramienta OLAP no soporta de la mejor forma este tipo de medidas, obteniendo información poco usable al necesitar una revisión bibliográfica.

En términos generales, la aceptación, al incluir medidas textuales con su función de agregación, es buena, ya que se obtiene información adicional a las tradicionales medidas numéricas, permitiendo un mejor análisis de los datos no estructurados

4.2.2.2 Análisis factorial

El Análisis Factorial es una técnica de reducción de datos [51]. En ocasiones las bases de datos están integradas por variables en las que aparece una amplia redundancia en la información, técnicamente se dice que son variables con un elevado nivel de inter-correlación. Ello plantea el problema de la multi-colinealidad que inutiliza la base para muchos modelos predictivos, por esto surge la necesidad de eliminar la redundancia informativa o eliminar la multi-colinealidad.

El Análisis Factorial permite si es necesario, sustituir el conjunto original de variables (preguntas) por otro sensiblemente menor en número de variables no observables o

hipotéticas, llamadas factores. Son definidas como variables incorreladas[21] (o con cierta correlación según el tipo de rotación aplicada) que explican los elevados niveles de inter-correlación presentes en la muestra. Estos factores, por tanto, eliminan la multi-colinealidad que describen las relaciones entre las variables. Dado el conjunto de variables inter-correladas el análisis factorial extrae un número de factores coincidente con el original de variables. Sin embargo, como éstas son internamente clasificadas por el método, la varianza global coincide con el número de variables. De esta varianza global cada factor recoge una cierta cantidad, es decir, explica una cierta proporción, cuanto mayor sea la cantidad explicada más importante es el factor.

Cuando los factores se conocen a priori y el diseño experimental se hace para obtener una puntuación para cada pregunta en los diferentes factores, el análisis recibe el nombre de Análisis Factorial Confirmatorio. Es decir, cuando se han planteado preguntas que están correlacionados con algunos factores, se pueden aplicar métodos estadísticos para determinar si los datos confirman su distribución, restricción o comportamiento.

Usando las respuestas de la encuesta mencionada en la sección anterior, los datos fueron examinados usando el análisis de componentes principales. Sin especificar el número de factores, tres factores representativos se generan con eigen valores[22] por encima de uno para cada una de las consultas explicando el 80% de la varianza de acuerdo al número de encuestas establecido. El factor uno es un eje que mide el tiempo de las consultas, el factor dos mide la facilidad de uso de las medidas textuales sobre las dimensiones y el factor tres mide la comprensión e interpretación de las medidas textuales en las consultas predefinidas. Para el análisis de los datos se establece la siguiente nomenclatura para las preguntas:

P1: El sistema presenta una fácil navegación?
P2: Es entendible la presentación de la medida textual para el análisis?
P3: El formato de presentación de la medida textual es adecuado?
P4: La información presentada por la medida textual satisface sus expectativas?,
P5: Es entendible la relación entre la medida textual y la(s) jerarquía(s) de la(s) dimensión(es)?
P6: El tiempo de respuesta fue lo suficientemente rápido?
P7: El tiempo de respuesta fue el esperado?
P8: El tiempo de respuesta de las consultas se ajusta a sus expectativas?
P9: En su percepción, la navegación en el modelo de documentos a través de las medidas textuales es útil en el caso de una revisión bibliográfica?

En la Tabla 3 se observa los tres componentes o factores generados en la matriz de componentes de cada una de las consultas. En cada celda se encuentra el valor de significancia de la pregunta para cada factor, de acuerdo a las cuatro consultas establecidas. Para este análisis hay que tener en cuenta los siguientes pasos:

1. El valor de significancia en cada celda debe ser superior a .50 ya que por debajo de este valor no brinda mayor información sobre los factores generados.

[21] Cuando el coeficiente de correlación es cero.
[22] Los eigen valores determinan el número de factores a partir de la varianza explicada en las preguntas encuestadas.

2. Se ubica la pregunta en el factor con el valor de significancia más alto para cada una de las consultas. Si la pregunta, de acuerdo a su semántica, no aplica en el factor establecido se procede a definirlo en los restantes factores teniendo en cuenta el paso 1.

De acuerdo a estos pasos se sombrea la celda correspondiente al valor de significancia de la pregunta en el factor respectivo.

	Consulta 1			Consulta 2			Consulta 3			Consulta 4		
	1	2	3	1	2	3	1	2	3	1	2	3
P1	-,059	-,719	-,083	-,127	-,007	,934	,165	,329	-,668	,394	-,368	-,499
P2	,211	,320	,803	,471	,588	-,525	,063	,758	-,301	-,188	,545	,604
P3	,639	-,250	,442	,590	-,276	,395	,573	,682	-,239	,608	,306	,504
P4	,659	,580	,245	,499	,698	,149	,189	,848	,352	,351	,853	-,230
P5	,062	,541	-,667	,118	,526	,586	-,261	-,093	,794	,099	,784	-,251
P6	,752	-,594	-,057	,743	-,638	,055	,884	-,421	-,046	,881	-,293	,080
P7	,893	-,154	-,258	,952	-,190	-,037	,937	-,072	,310	,966	-,116	,023
P8	,921	-,196	-,204	,952	-,190	-,037	,937	-,072	,310	,954	-,113	,079
P9	,673	,591	-,172	,417	,746	,044	-,236	,721	,571	,201	,813	-,294

Tabla 3. Matriz de Componentes para las consultas (C1, C2, C3, C4)

Esto permitió establecer tres factores de análisis para evaluar el modelo multidimensional en la herramienta OLAP sin importar el tipo de consulta que se haga, quedando los factores y las preguntas organizados de la siguiente manera (ver Figura 58).

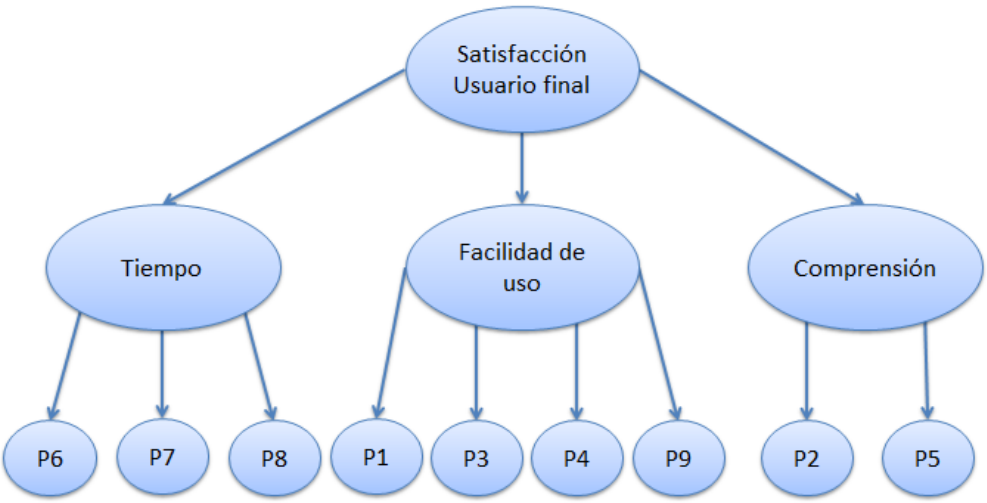


Figura 58. Modelo para medir la satisfacción de usuario

4.2.2.3 Análisis Multivariado

El análisis multivariado [52] se define como un conjunto de métodos que permite el análisis simultaneo de más de dos variables observadas en una investigación,

permitiendo el estudio de la información multivariante en el sentido de que hay varias variables medidas para cada individuo u objeto estudiado.

El análisis de componentes principales es una técnica multivariada que permite seccionar la información contenida en un conjunto de variables de interés en nuevas variables independientes, es decir, este análisis está centrado en la reducción de la dimensión del espacio de los datos, pero también es empleado para realizar un seguimiento sobre los componentes principales obtenidos para comprobar hipótesis establecidas en un estudio de análisis multivariado para identificar datos atípicos en el conjunto de datos.

Con base en lo anterior y al haber definido un conjunto de preguntas en una serie de factores se desea, a través de este análisis, determinar la contribución de los factores encuestados a medida que los usuarios interactúan con las medidas textuales en la herramienta OLAP.

Para este análisis se tuvo en cuenta los factores principales encuestados, facilidad de uso y tiempo de consulta, y la matriz de componentes obtenido en cada una de las consultas (ver Tabla 3), permitiendo ubicar a cada encuestado en un espacio bidimensional con el fin de observar su comportamiento y compararlos en cada uno de los factores a partir de los resultados de la encuesta.

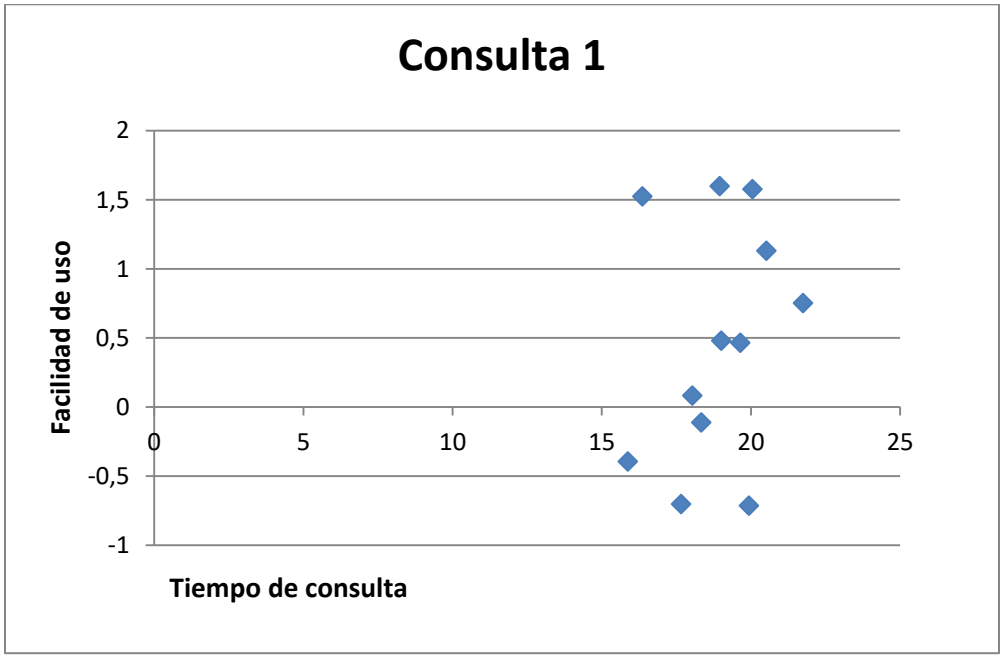


Figura 59. Dispersión de los encuestados en la consulta 1

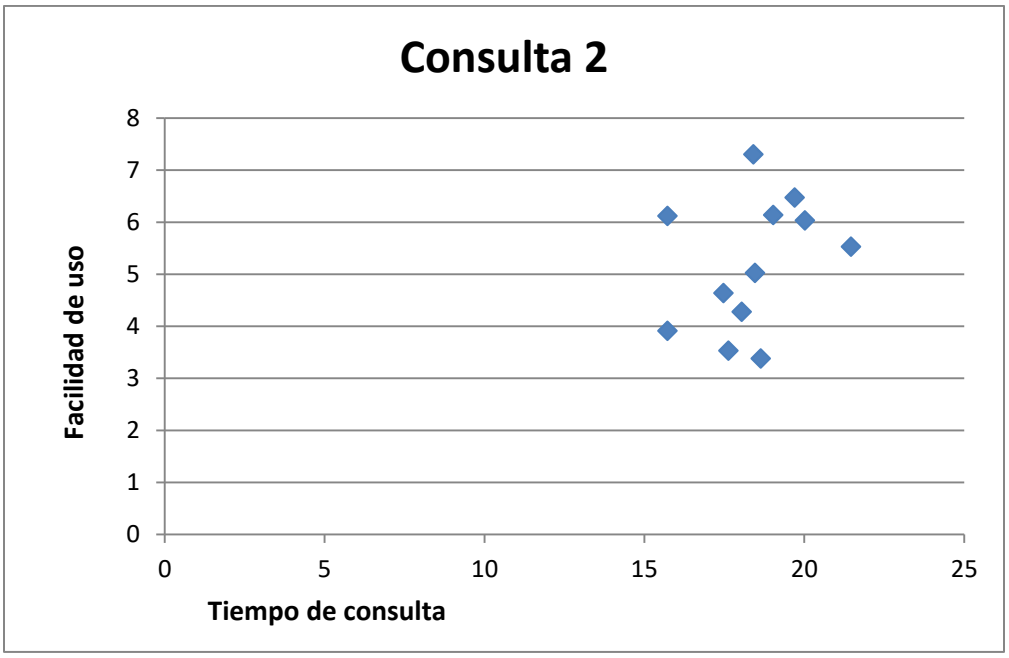


Figura 60. Dispersión de los encuestados en la consulta 2

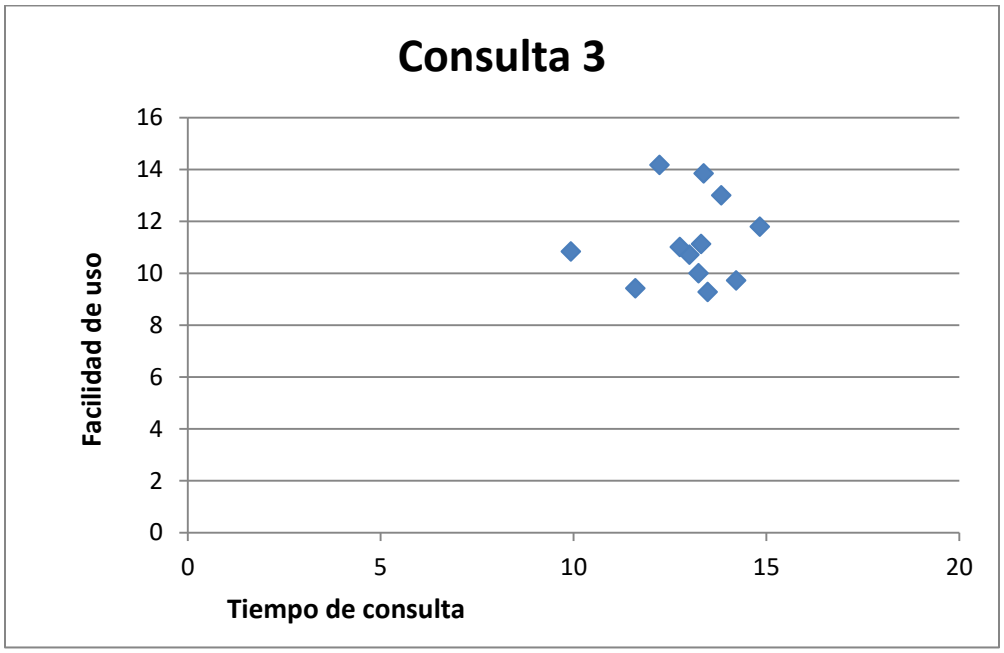


Figura 61. Dispersión de los encuestados en la consulta 3

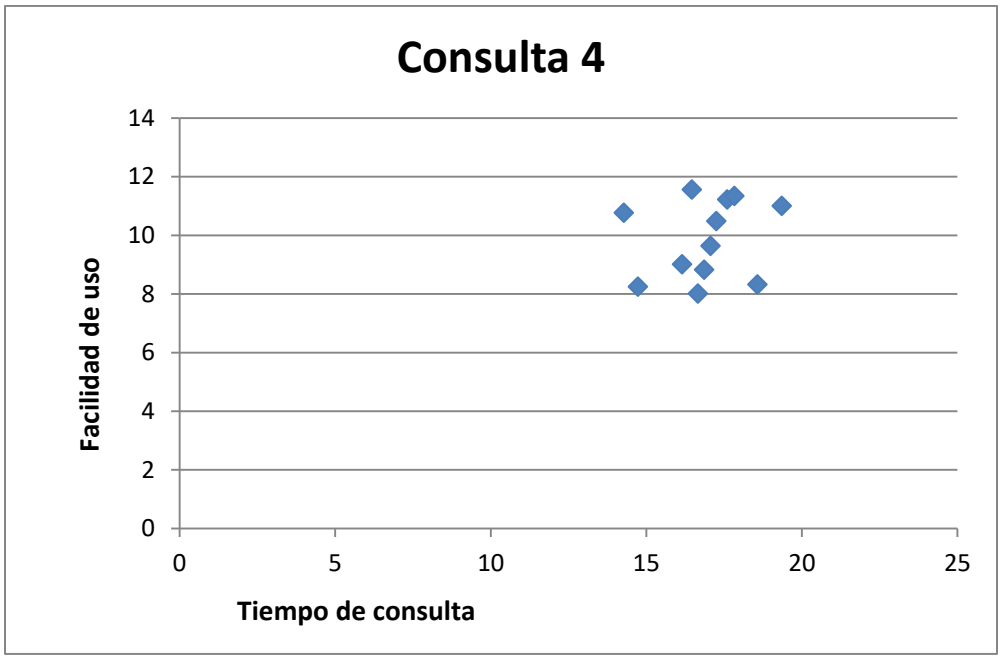


Figura 62. Dispersión de los encuestados en la consulta 4

Como se observa en las gráficas anteriores, para el factor tiempo de consulta, la dispersión en la opinión de los encuestados es homogénea ya que conserva en los cuatro momentos el mismo rango de puntuación, es decir el tiempo con que se ejecuta las consultas a medida que se interactúa con las medidas textuales en la herramienta OLAP es buena para los encuestados.

El factor facilidad de uso tiene un comportamiento especial, como se observa en la Figura 59 hay una mayor dispersión en la opinión de los encuestados donde la puntuación fluctúa entre -1 y 2, esto se debe a que se incluyen medidas atípicas a las tradicionales medidas numéricas que se definen en un modelo de bodega de datos. Pero a medida que el encuestado interactúa con las demás consultas, esta opinión va mejorando, observándose un proceso en la aceptación y navegación al interactuar con este tipo de medidas textuales sobre una herramienta OLAP.

5. CONCLUSIONES

- Se definió e implementó un modelo multidimensional que integra datos no estructurados en una bodega de documentos, donde el nivel de granularidad es por documento detallando la información relevante de cada documento, la navegación del modelo es apoyado por una jerarquía de tópicos generada automáticamente por el algoritmo IGBHSK y la sumarización (resumen) de los documentos se haga mediante las medidas textuales con su función de agregación incorporadas en el modelo.

- Se observó que al interactuar con medidas textuales sobre una herramienta OLAP clásica, permiten obtener mayor información para la toma de decisiones en las organizaciones a las típicas medidas numéricas. Teniendo en cuenta que las herramientas OLAP actuales aun no dan soporte para el buen manejo de este tipo de medidas.

- Se definieron e implementaron tres medidas textuales, necesarias para el análisis de los datos, en este caso, documentos de texto. Logrando que a través de las consultas se obtenga la información relevante de un documento a partir de la diferentes dimensionalidades asociadas al modelo propuesto.

- Al definir una granularidad a nivel de documento, conlleva a modelar e implementar relaciones M-M entre la tabla de hechos principal y las dimensiones, que permitan de manera óptima almacenar la información, para este caso, de los documentos científicos utilizados.

- Al implementar medidas textuales, cuya definición no es tan simple como para las medidas numéricas, es necesario definir estas medidas con el lenguaje de consultas MDX y su función de agregación mediante un procedimiento almacenado en un lenguaje de programación estándar.

- El modelo multidimensional propuesto sirve como guía para ser implementado en un ambiente empresarial, en el cual, se almacene información de tópicos importantes en los documentos que se manejan en la empresa y sirva como apoyo a la toma de decisiones. Además el modelo multidimensional permite de forma relativamente sencilla adicionar nuevas dimensiones que aporten información para el análisis de los datos de acuerdo a las necesidades de la empresa.

- Con respecto a la evaluación de las medidas textuales por parte de los usuarios finales, se observó que existe un problema cuando se muestran las medidas textuales en una herramienta OLAP, debido a que estas no presentan un buen soporte para poder visualizar medidas textuales.

- A pesar de que el tiempo de respuesta de las medidas textuales incrementa a medida que se consulta mayor cantidad de documentos y dependiendo del tipo de consulta (con una dimensión, acoplada o por matriz), las consultas realizadas entre los diferentes usuarios permitieron establecer un nivel de aceptación bueno respecto al tiempo de respuesta de la aplicación con diferentes medidas textuales, además que

les permitió tener una visión más amplia de la información en contraste a las medidas numéricas tradicionales.

- Los algoritmos del modelo multidimensional utilizados para la generación de la jerarquía de tópicos (IGBHSK modificado) y de las probabilidades (PLSA), pueden ser reemplazados por otros algoritmos de acuerdo a las necesidades que se tengan, teniendo en cuenta que: primero, el algoritmo que se use para generar la jerarquía de tópicos debe agrupar un conjunto de documentos, permitir solapamiento entre grupos y etiquetar cada grupo. Segundo, si la medida probabilística propuesta es otra, verificar que el nuevo algoritmo asocie un valor numérico entre los documentos y la jerarquía de tópicos, y los tópicos con los términos de los documentos. Al tener en cuenta estos dos aspectos, como lo fue para este proyecto, se lograra incorporar este tipo de datos no estructurados en una bodega de documentos.

6. TRABAJO FUTURO

Realizar más evaluación del modelo multidimensional, con un mayor número de documentos (en lo posible millones de registros), que permita identificar una tendencia en cuanto al tiempo de respuesta de las consultas que involucran el uso de medidas textuales.

Diseñar e implementar un control o librería para una herramienta OLAP que se adecue a las necesidades de este tipo de medidas textuales, mejorando la visualización de la salida de los datos al usuario final.

Incluir el manejo de cargas incrementales de los documentos en el proceso ETL, que permita la actualización de la información en la base de datos relacional, es decir, que cada vez que se desee añadir un nuevo documento, solo se realice los cuatro procesos de la arquitectura general para los nuevos documentos y no para todo el conjunto de documentos. Esto permitiría reducir el tiempo de procesamiento, clasificación y ponderación de los nuevos documentos.

Adicionar al algoritmo IGBHSK modificado un proceso de Lematización multilenguaje que permita el análisis de los documentos en múltiples idiomas, debido a que el proceso existente es adecuado solo para documentos escritos en lenguaje inglés.

Adicionar a las medidas de IR probabilísticas una escala de evaluación que permita al usuario final conocer la importancia y relevancia de los documentos en los tópicos o dimensiones cuando se realiza una consulta, obteniendo mayor rango de posibilidades en escoger los documentos de interés.

Para mejorar los resultados en la encuesta, es necesario que se disponga del código fuente de la herramienta OLAP, o proponer una, que mejore el manejo de las medidas textuales, permitiendo al usuario final, a través de las consultas, observar las medidas en forma de lista o personalizables, donde el ranking de las probabilidades se muestren en gráficos típicas de las herramientas OLAP comprensibles para el usuario final.

Al incorporar este modelo multidimensional en un ambiente empresarial se deben tener en cuenta los siguientes aspectos:

- El modelo debe contemplar siempre las tablas de dimensión documento, tópico y palabra, junto con las medidas probabilísticas que se definen en el modelo generadas por PLSA.

- Las tablas de dimensión diferentes a las antes mencionadas, pueden cambiar, dependiendo de las necesidades de la empresa o la lógica del negocio que se maneje. Por ejemplo la dimensión autor, si la lógica del negocio está centrada en las ventas, podría ser dimensión vendedor con atributos diferentes, el cual hace reportes de las ventas en una determinada área de la organización.

- Si se usa una metodología para el modelamiento de bodega de datos (Ej. La metodología propuesta por Ralph Kimball) y se desea incorporar datos no estructurados, hay que tener en cuenta los aspectos anteriores y las particularidades del modelo propuesto.

- Tener en cuenta que la herramienta OLAP que se haya escogido, brinde el soporte para las medidas textuales creadas, permitiendo que la visualización y el análisis de los datos no estructurados sea comprensible y adecuado para el usuario final objetivo.

7. REFERENCIAS BIBLIOGRÁFICAS

1. M, S.G., *Los datos: materia prima y real valor de las organizaciones, El verdadero impacto de las bodegas de datos en las organizaciones*, in *ACIS*. 2008, Revista Sistemas.
2. Dan, S., *Document Warehousing and Text Mining: Techniques for Improving Business Operations, Marketing, and Sales*. 2001: John Wiley & Sons, Inc. 560.
3. Yuanyuan, T., A.H. Richard, and M.P. Jignesh, *Efficient aggregation for graph summarization*, in *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*. 2008, ACM: Vancouver, Canada. p. 567–580.
4. Ježek, K. and J. Steinberger. *Automatic Text Summarization (The state of the art 2007 and new challenges)*. in *Znalosti 2008*. 2008. Bratislava, Slovakia.
5. Steinberger, J. and K. Ježek, *Text Summarization: An Old Challenge and New Approaches*, in *Foundations of Computational, IntelligenceVolume 6*. 2009. p. 127-149.
6. Duo, Z., et al., *Topic modeling for OLAP on multidimensional text databases: topic cube and its applications.* Stat. Anal. Data Min., 2009. **2**(5, 6): p. 378-395.
7. Cindy Xide, L., et al., *Text Cube: Computing IR Measures for Multidimensional Text Database Analysis*, in *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining*. 2008, IEEE Computer Society.
8. C. Cobos, J.A., W. Constain, M. Mendoza, and E. León, *Web document clustering based on Global-Best Harmony Search, K-means, Frequent Term Sets and Bayesian Information Criterion*, in *IEEE Congress on Evolutionary Computation (IEEE CEC), Barcelona, Spain, 2010*. 2010. p. 4637-4644.
9. Thomas, H., *Probabilistic latent semantic indexing*, in *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*. 1999, ACM: Berkeley, California, United States.
10. William, J.D. and T. Gholamreza, *The measurement of end-user computing satisfaction.* MIS Q., 1988. **12**(2): p. 259-274.
11. Serrano, C.E., *Modelo Integral para el Profesional en Ingeniería*. 2005, Popayán: Editorial Universidad del Cauca.
12. Kimball, R., *The data warehouse lifecycle toolkit: expert methods for designing, developing, and deploying data warehouses*. 1998: Wiley.
13. Inmon, W.H., *Building the data warehouse*. 2005: Wiley. 576.
14. Kimball, R., *Is ER modeling hazardous to DSS?*, in *DBMS Magazine*. october 1995.
15. Ladjel, B., K. Kamalakar, and M. Mukesh, *Some issues in design of data warehousing systems*, in *Data warehousing and web engineering*. 2002, IRM Press. p. 22-76.
16. Gill, H.S. and P.C. Rao, *DATA WAREHOUSING : LA INTEGRACION DE INFORMACION PARA LA MEJOR TOMA DE DECISIONES*. 1996: Prentice Hall Hispanoamericana.
17. Ralph, K. and R. Margy, *The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling*. 2002: John Wiley & Sons, Inc. 416.
18. Romero, O. and A. Abelló, *A Survey of Multidimensional Modeling Methodologies.* International Journal of Data Warehousing and Mining, 2009. **5**(2).
19. Ralph, K., et al., *The Data Warehouse Lifecycle Toolkit: Expert Methods for Designing, Developing and Deploying Data Warehouses with CD Rom*. 1998: John Wiley; Sons, Inc. 800.

20. Datta, A. and H. Thomas, *The cube data model: a conceptual model and algebra for on-line analytical processing in data warehouses.* Decis. Support Syst., 1999. **27**(3): p. 289-301.
21. Cody, W.F., et al., *The integration of business intelligence and knowledge management.* IBM Syst. J., 2002. **41**(4): p. 697-713.
22. McCabe, M.C., et al., *On the design and evaluation of a multi-dimensional approach to information retrieval (poster session)*, in *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*. 2000, ACM: Athens, Greece.
23. Tseng, F.S.C. and A.Y.H. Chou, *The concept of document warehousing for multi-dimensional modeling of textual-based business intelligence.* Decis. Support Syst., 2006. **42**(2): p. 727-744.
24. Franck, R., et al., *Top_Keyword: An Aggregation Function for Textual Document OLAP*, in *Proceedings of the 10th international conference on Data Warehousing and Knowledge Discovery*. 2008, Springer-Verlag: Turin, Italy.
25. Alkis, S., et al., *Multidimensional content eXploration.* Proc. VLDB Endow., 2008. **1**(1): p. 660-671.
26. Lin, C.X., et al. *Text Cube: Computing IR Measures for Multidimensional Text Database Analysis*. in *Data Mining, 2008. ICDM '08. Eighth IEEE International Conference on*. 2008.
27. Manning, C., P. Raghavan, and H. Schütze, *Introduction to Information Retrieval*. 2008, Cambridge University Press: Cambridge, England.
28. Baeza-Yates, R., A. and B. Ribeiro-Neto, *Modern Information Retrieval*. 1999: Addison-Wesley Longman Publishing Co., Inc. 513.
29. Zhang, D., C. Zhai, and J. Han. *Topic Cube: Topic Modeling for OLAP on Multidimensional Text Databases*. in *SIAM International Conference on Data Mining*. 2009: Society for Industrial and Applied Mathematics.
30. Zhang, D., et al., *Topic modeling for OLAP on multidimensional text databases: topic cube and its applications.* Stat. Anal. Data Min., 2009. **2**(5\‐6): p. 378-395.
31. Zdravko, M. and T.L. Daniel, *Data Mining the Web: Uncovering Patterns in Web Content, Structure, and Usage*. 2007: Wiley-Interscience.
32. Yintao, Y., et al., *iNextCube: information network-enhanced text cube.* Proc. VLDB Endow., 2009. **2**(2): p. 1622-1625.
33. Yizhou, S., et al., *RankClus: integrating clustering with ranking for heterogeneous information network analysis*, in *Proceedings of the 12th International Conference on Extending Database Technology: Advances in Database Technology*. 2009, ACM: Saint Petersburg, Russia.
34. Yizhou, S., Y. Yintao, and H. Jiawei, *Ranking-based clustering of heterogeneous information networks with star network schema*, in *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*. 2009, ACM: Paris, France.
35. Thomas, H., *Unsupervised Learning by Probabilistic Latent Semantic Analysis.* Mach. Learn., 2001. **42**(1-2): p. 177-196.
36. S. Deerwester, S.T.D., G. W. Furnas, Landauer. and a.R.H. T. K., *Indexing by latent semantic analysis.* Journal of the American Society for Information Science, 1990: p. 41.
37. Quesada, J. *Latent problem solving analysis (LPSA): A computational theory of representation in complex, dynamic problem solving tasks*. in *24th Annual*

Conference of the Cognitive Science Society. 2003. Fairfax, VA. Lawrence Erlbaum Associates , Mahwah, NJ.
38. A.P. Dempster, N.M.L., and D.B. Rubin, *Maximum likelihood from incomplete data via the EM algorithm.* Journal of the Royal Statistical Society, 1977. **39**: p. 1-38.
39. Ralambondrainy, H., *A conceptual version of the K-means algorithm.* Pattern Recogn. Lett., 1995. **16**(11): p. 1147-1157.
40. Osiński, S., *Lingo: An Algorithm for clustering of web search results*, in *Master Poland: Poznań University of Technology*. 2003. p. 91.
41. Paul, C., et al., *Documenting Software Architectures: Views and Beyond*. 2002: Pearson Education. 512.
42. Mary, S. and G. David, *Software architecture: perspectives on an emerging discipline*. 1996: Prentice-Hall, Inc. 242.
43. Markov, Z. and D.T. Larose, *Data mining the Web: uncovering patterns in Web content, structure, and usage*. 2007: Wiley-Interscience.
44. Whitehorn, M., R. Zare, and M. Pasumansky, *Fast track to MDX*. 2006: Springer.
45. Smith, B.C., C.R. Clay, and C. Hitachi, *Microsoft SQL Server 2008 MDX Step by Step*. 2009: Microsoft Press. 400.
46. Irina, G., B. Alexander, and M. Edward, *Microsoft SQL Server 2008 Analysis Services Unleashed*. 2008: SAMS. 888.
47. Salton, G., *Automatic text processing: the transformation, analysis, and retrieval of information by computer*. 1989: Addison-Wesley.
48. Gospodnetic, O., Hatcher E. and Cutting D. and E. Hatcher, *Lucene in action*. 2005, Michigan: Manning. 421.
49. Lucene.NET. *Sitio web de Lucene.NET.* 2010 [cited; Available from: http://incubator.apache.org/lucene.net/.
50. Cheng, H., *Parallel Implementations Of Probabilistic Latent Semantic Analysis On Graphic Processing Units*. 2010, University of Illinois at Urbana-Champaign.
51. Poza Lara, C., *Técnicas estadísticas multivariantes para la generación de variables latentes*, in *Revista-Escuela de Administración de Negocios*. septiembre-diciembre, 2008: Universidad EAN Colombia.
52. Peña, D., *Análisis de datos multivariantes*. 2002: MCGRAW HILL. 539.

ANEXO A – CONFIGURACIÓN DEL MODELO MULTIDIMENSIONAL EN LA HERRAMIENTA ANALYSIS SERVICES

Configuración proyecto Analysis Services para el modelo Multidimensional propuesto

Para crear un proyecto de Analysis Services es necesario contar con el SQL Sever Business Intelligence Development Studio 2008 (ver **Figura 63**), el cual contiene Analysis Services 2008 necesario para crear el modelo lógico, Integration Services 2008 necesario para el proceso ETL, y Reporting Services el cual genera reportes del cubo creado.

Primero: Se ejecuta SQL Sever Business Intelligence Development Studio 2008.

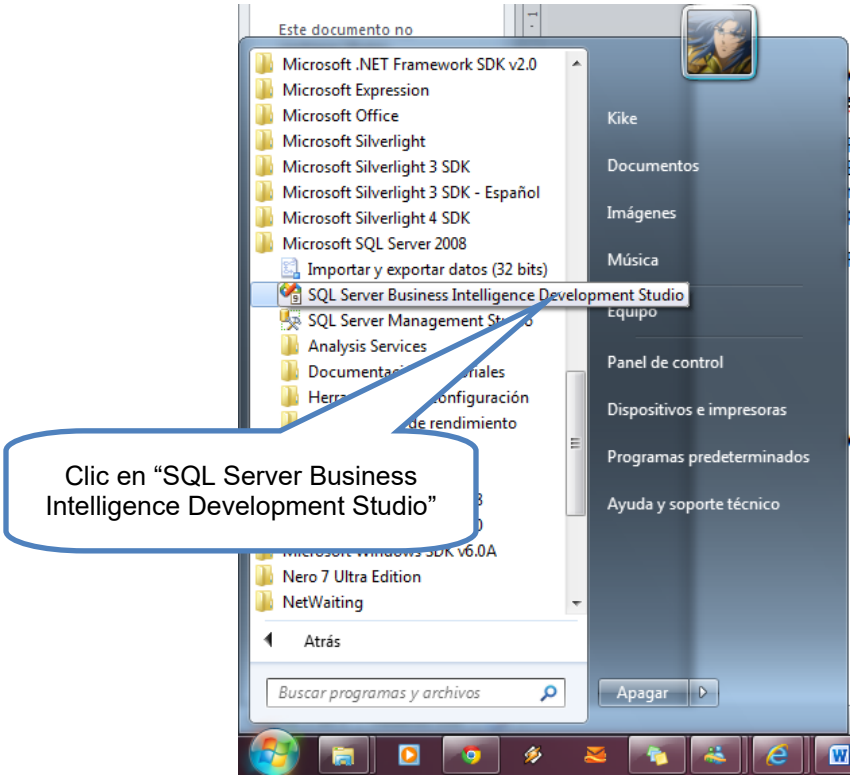


Figura 63. Programa SQL Server Intelligence Development Studio

Segundo: Se crea un nuevo proyecto (ver **Figura 64**) de Analysis Services (Archivo – Nuevo – Proyecto…)

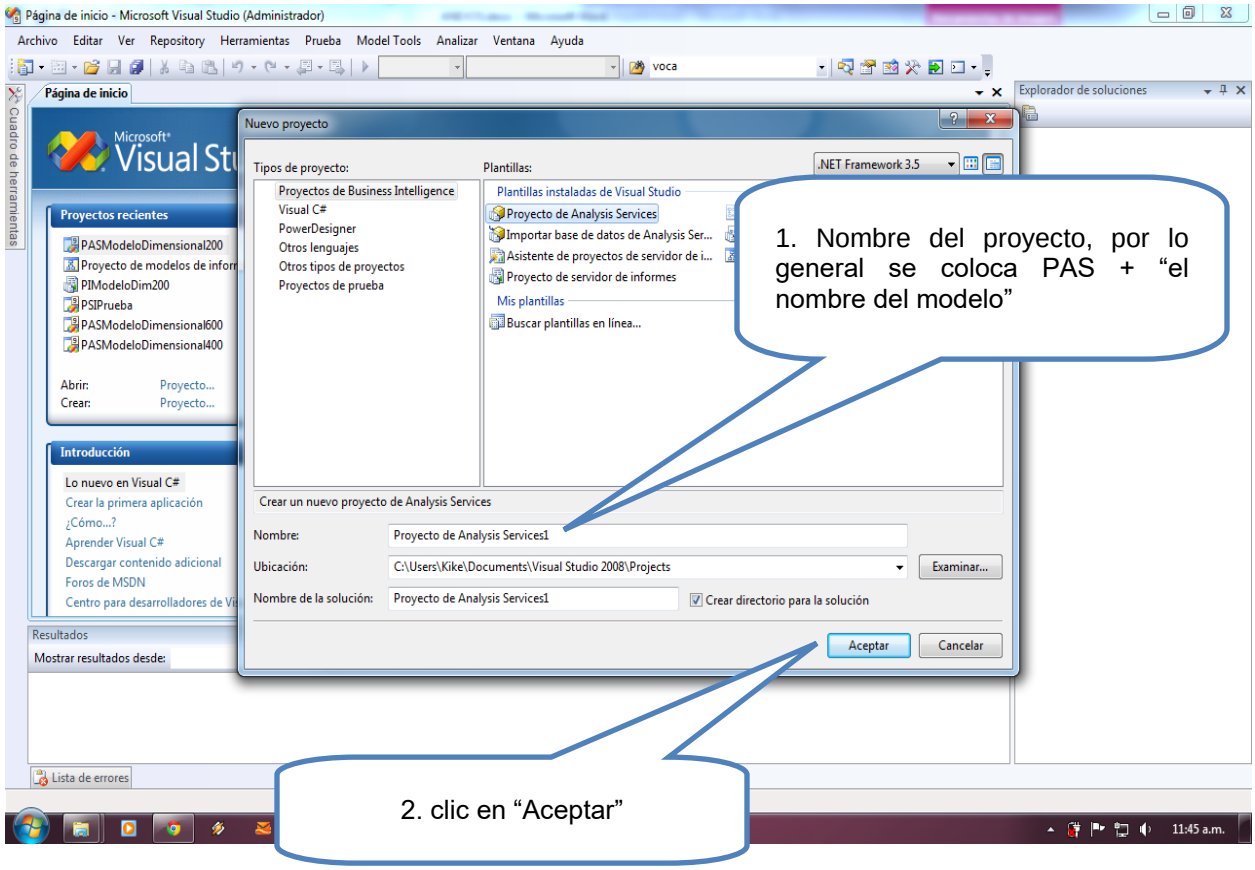


Figura 64. Nuevo proyecto Analysis services

Tercero: Se configura el origen de los datos (ver **Figura 65**). Para este caso una base de datos relacional creada en SQL server 2008.

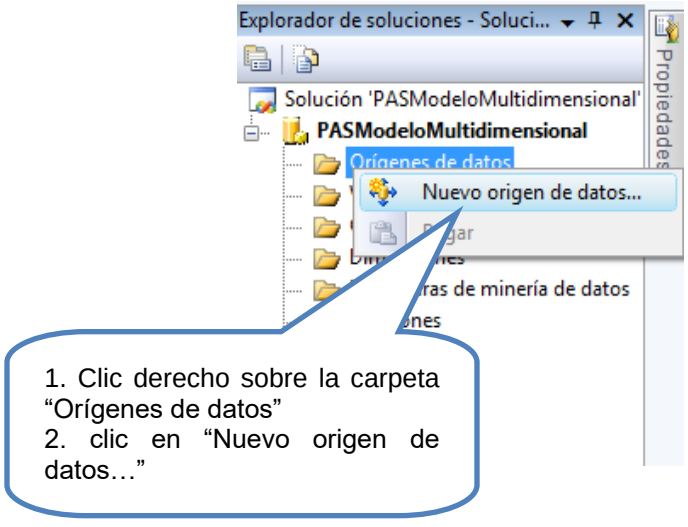


Figura 65. Nuevo origen de datos para el cubo

Cuarto: Se selecciona la fuente donde se encuentra la base de datos relacional del modelo multidimensional mediante el asistente y finalización del proceso de configuración (ver Figura 66, Figura 67, Figura 68, Figura 69 y Figura 70).

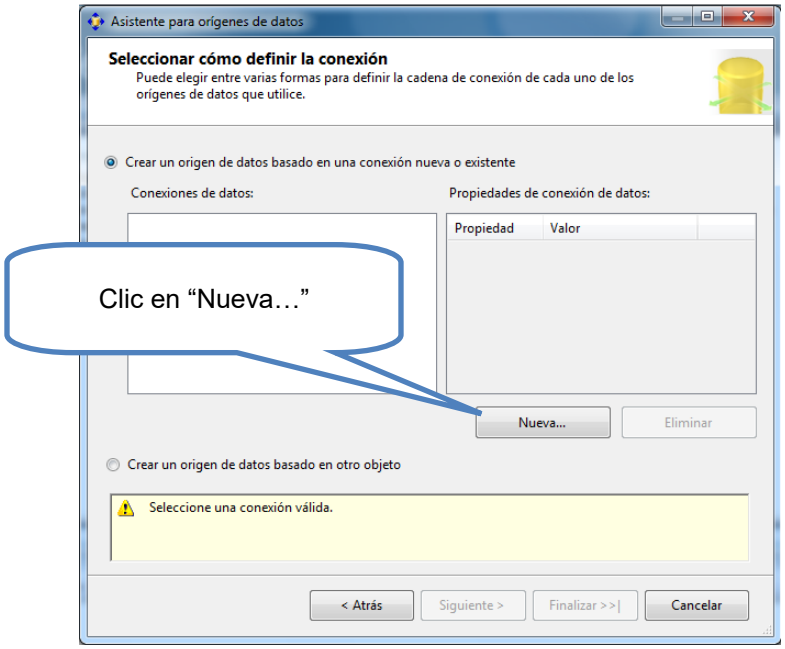


Figura 66. Crear nuevo conexión

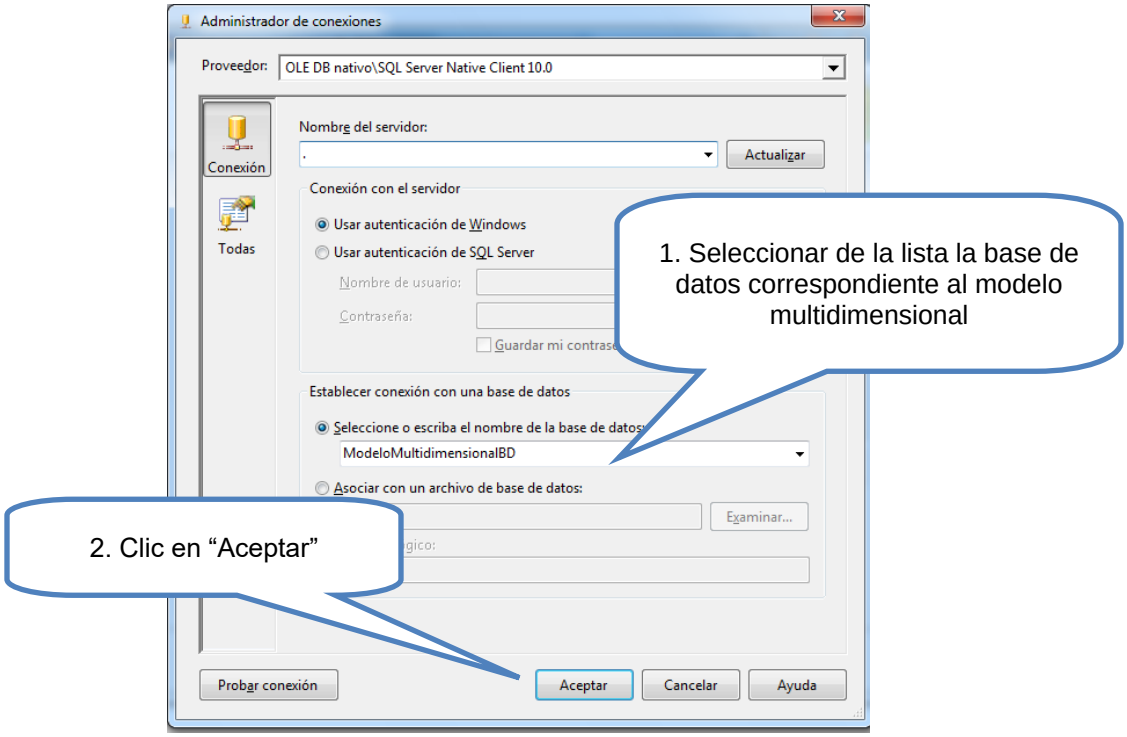


Figura 67. Seleccionar la base de datos relacional del modelo multidimensional

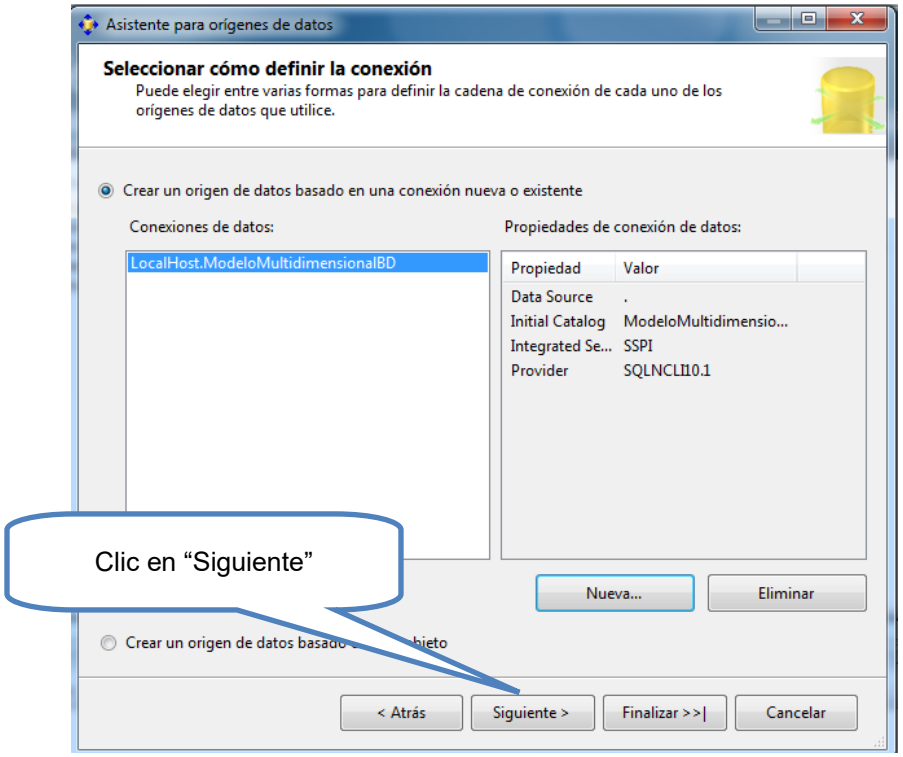


Figura 68. Configuración de la base de datos relacional confirmada.

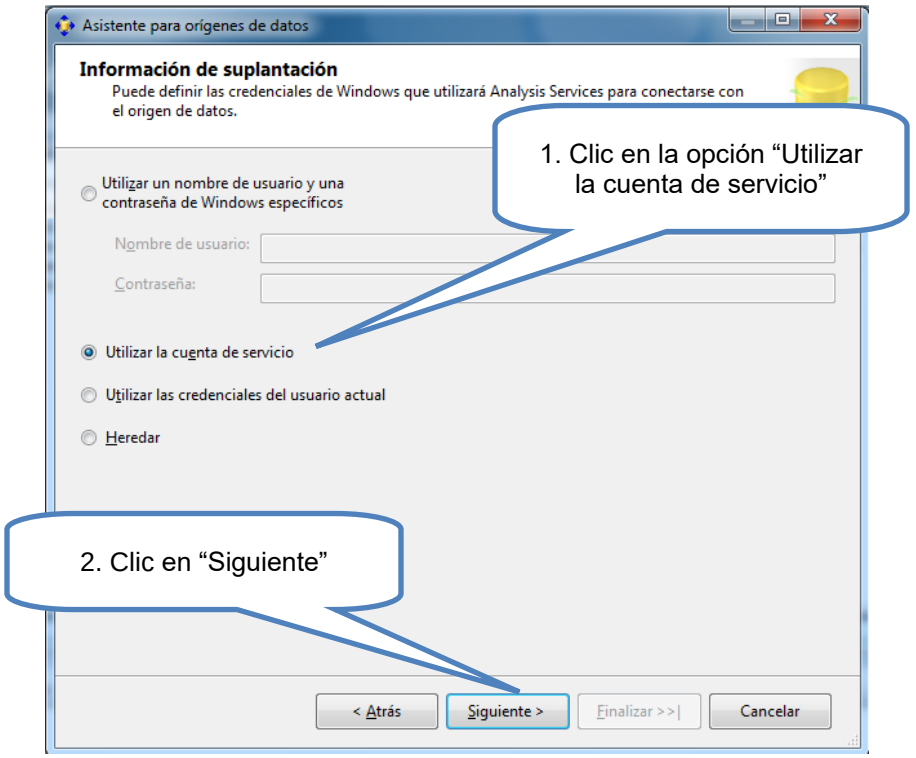


Figura 69. Seleccionar la opción “Utilizar la cuenta de servicio”

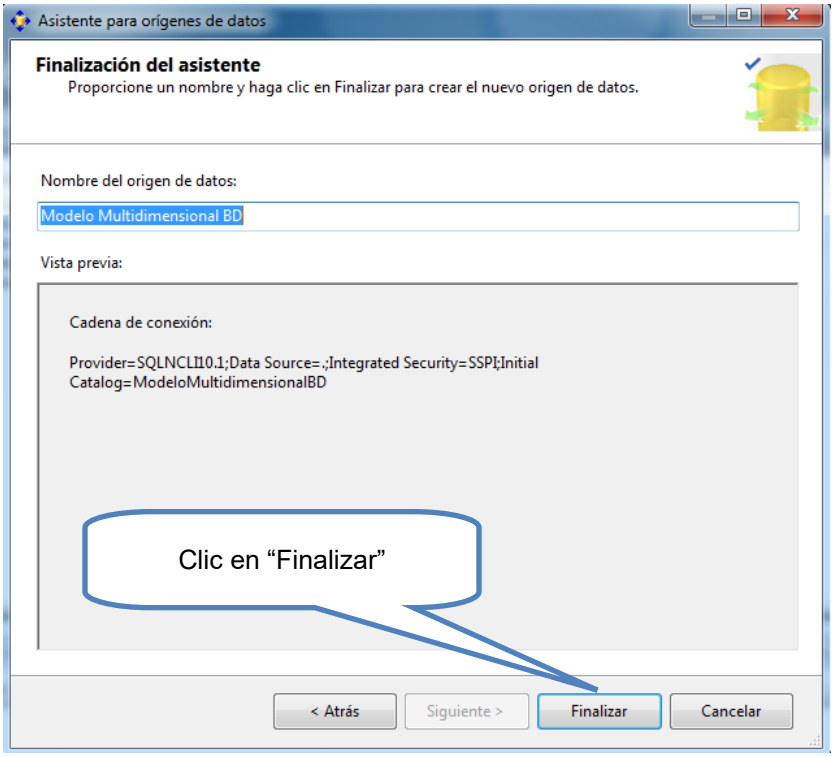


Figura 70. Finalización del asistente del origen de datos

Quinto: Se crea una nueva vista de origen de datos (ver **Figura 71**, **Figura 72**, **Figura 73**, **Figura 74** y **Figura 75**). Esta vista es una representación lógica de los datos que usan los objetos Analysis Services y se genera a partir de los orígenes de datos ya definidos en la base de datos relacional.

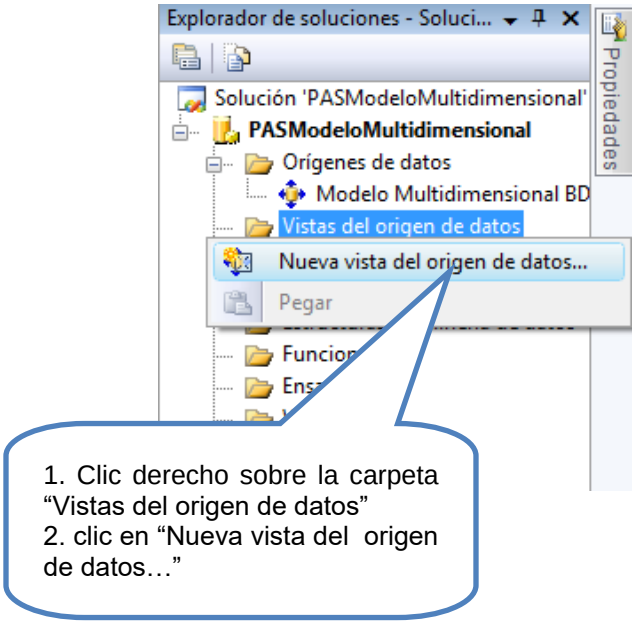


Figura 71. Nueva vista del origen de datos para el cubo

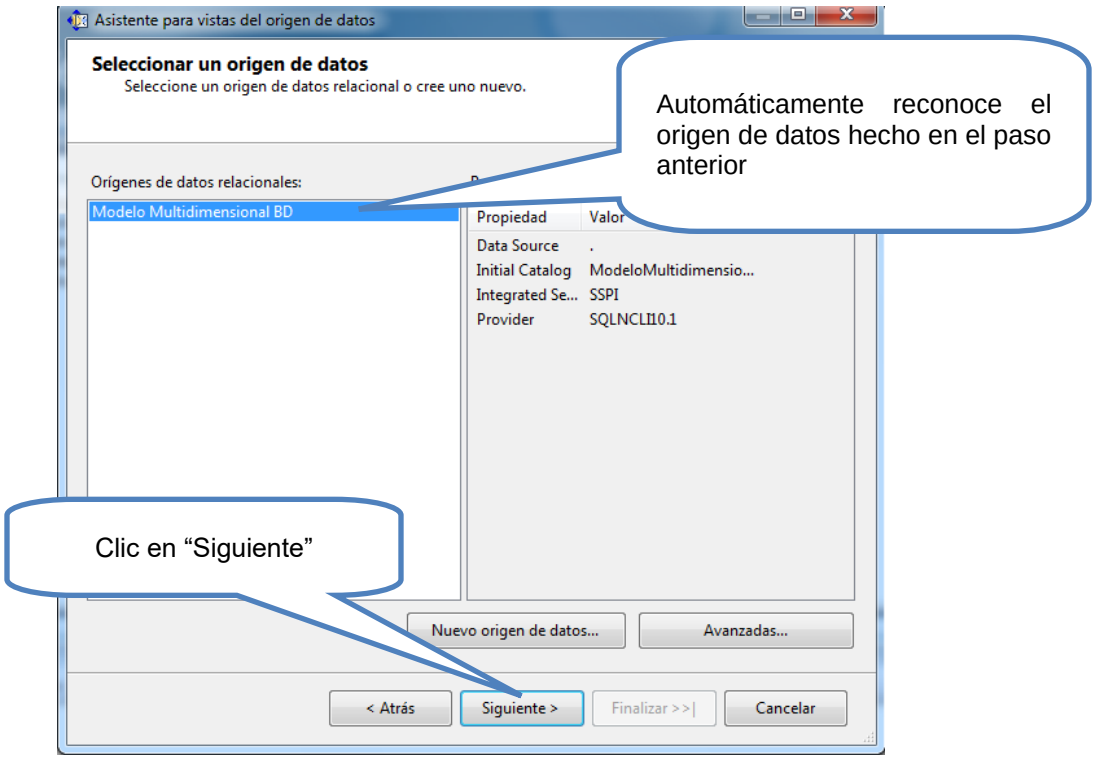


Figura 72. Verificación el origen de datos

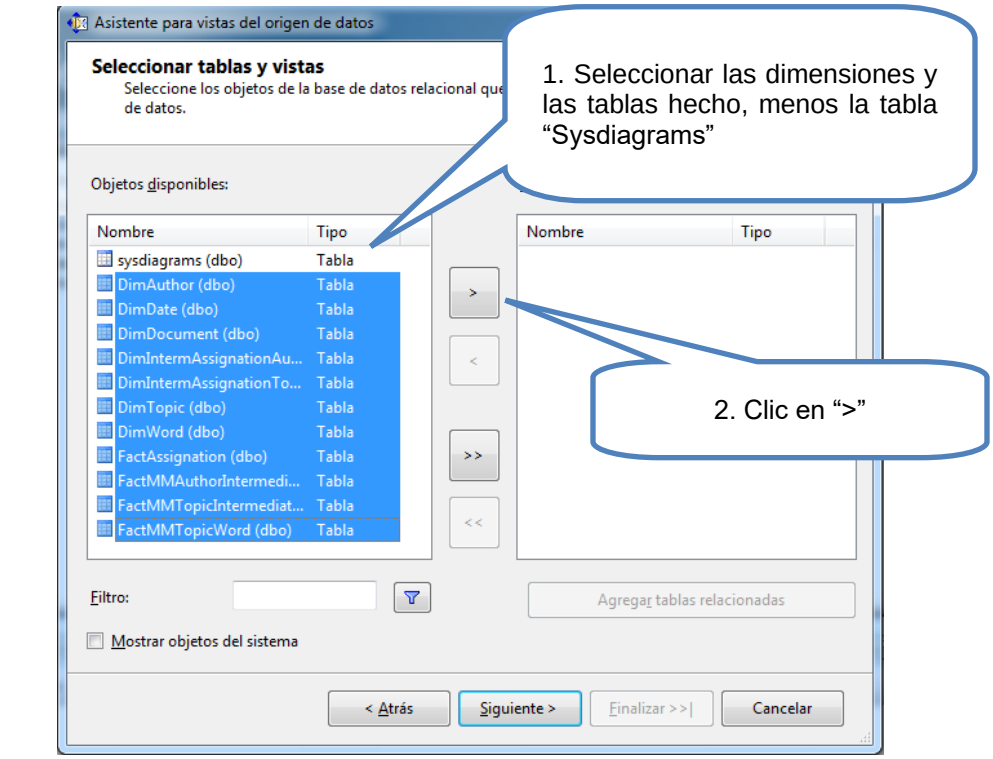


Figura 73. Seleccionar las tablas de dimensión y de hecho en el asistente

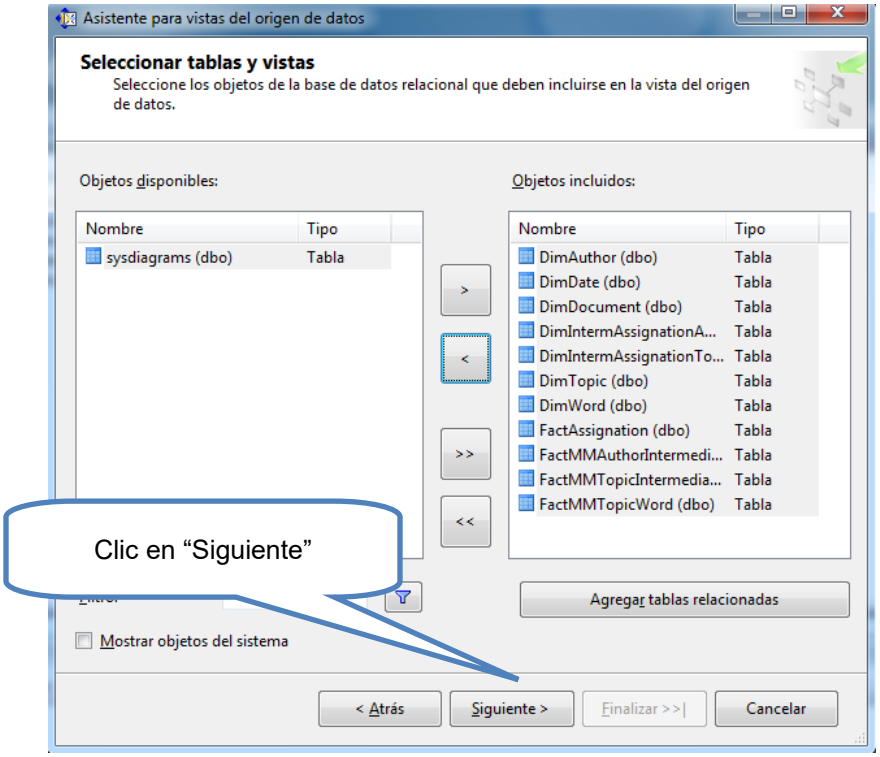


Figura 74. Confirmar las tablas de dimensión y de hecho, y continuar con el asistente

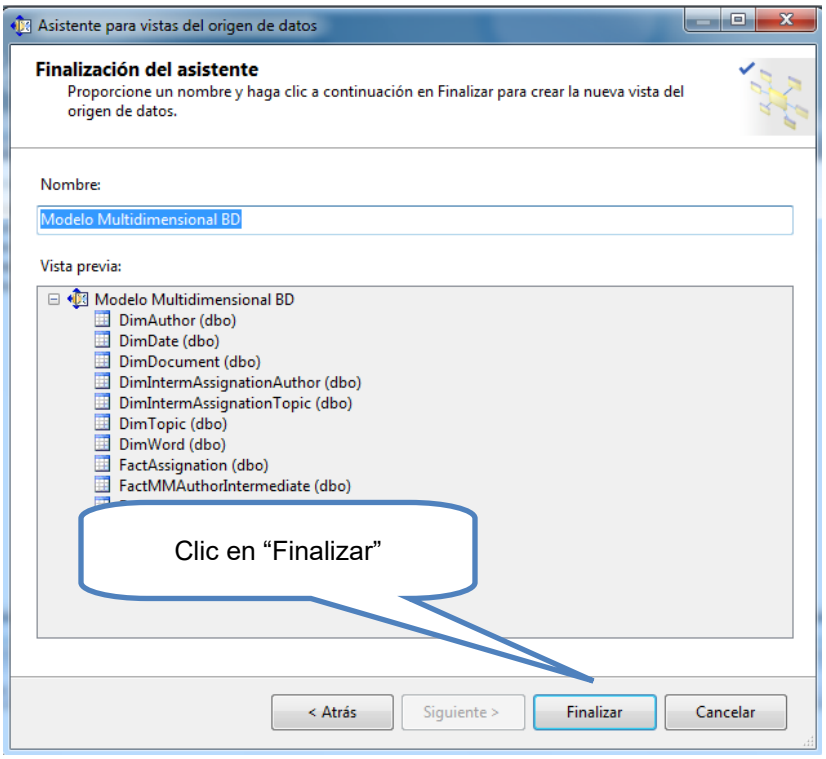


Figura 75. Finalización del asistente de la vista de origen de datos

Sexto: Se crea un nuevo cubo (ver **Figura 76**, **Figura 77**, **Figura 78**, **Figura 79**, **Figura 80** y **Figura 81**). Al definir un cubo también se definen los grupos de medida y las dimensiones del cubo.

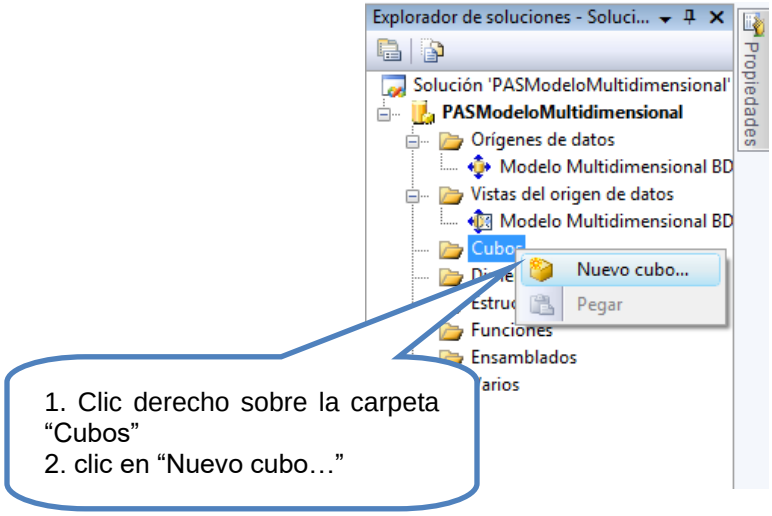


Figura 76. Nuevo cubo

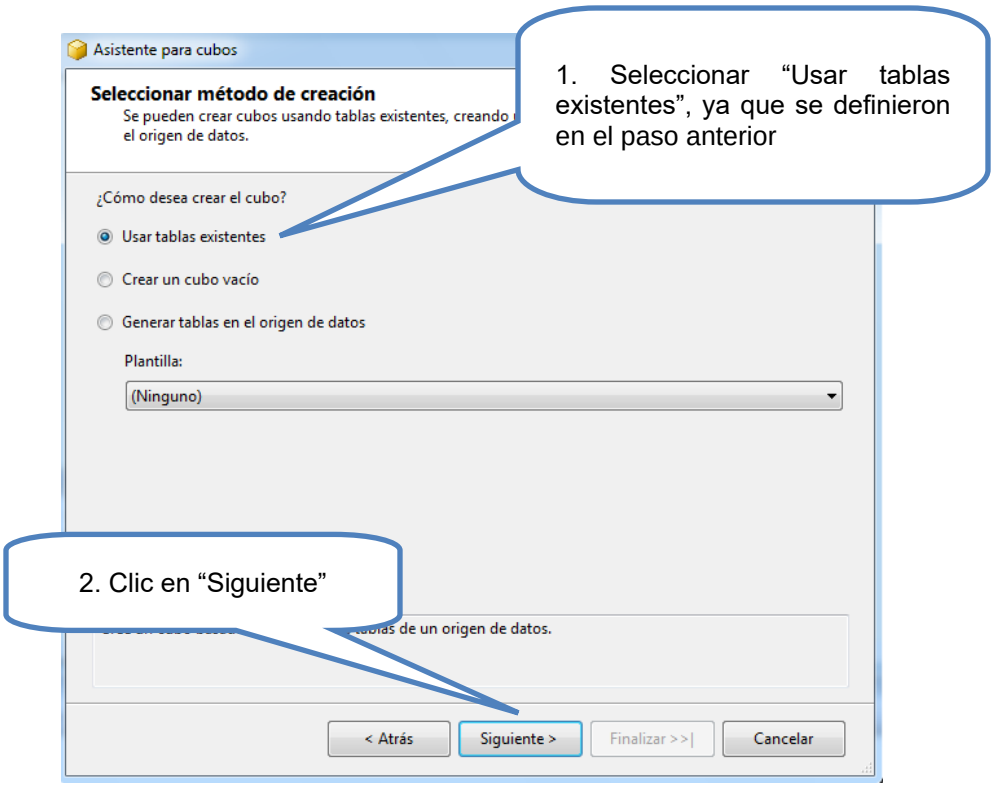


Figura 77. Seleccionar la opción "Usar tablas existentes"

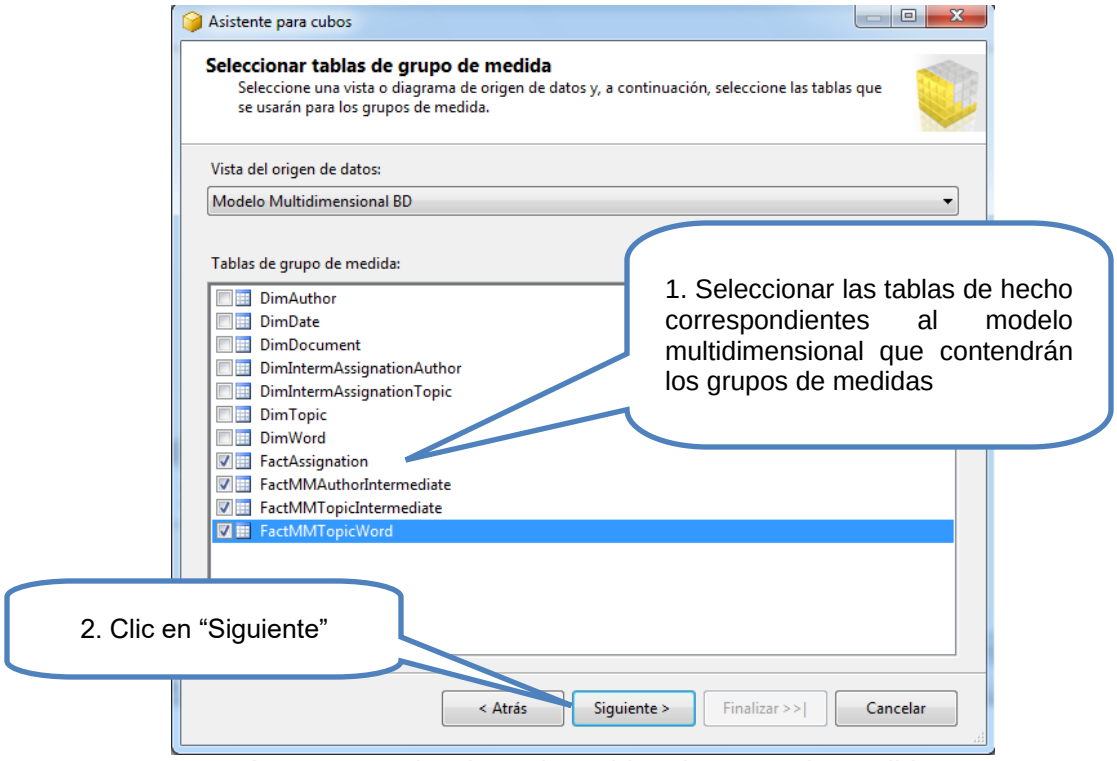


Figura 78. Seleccionar las tablas de grupo de medida

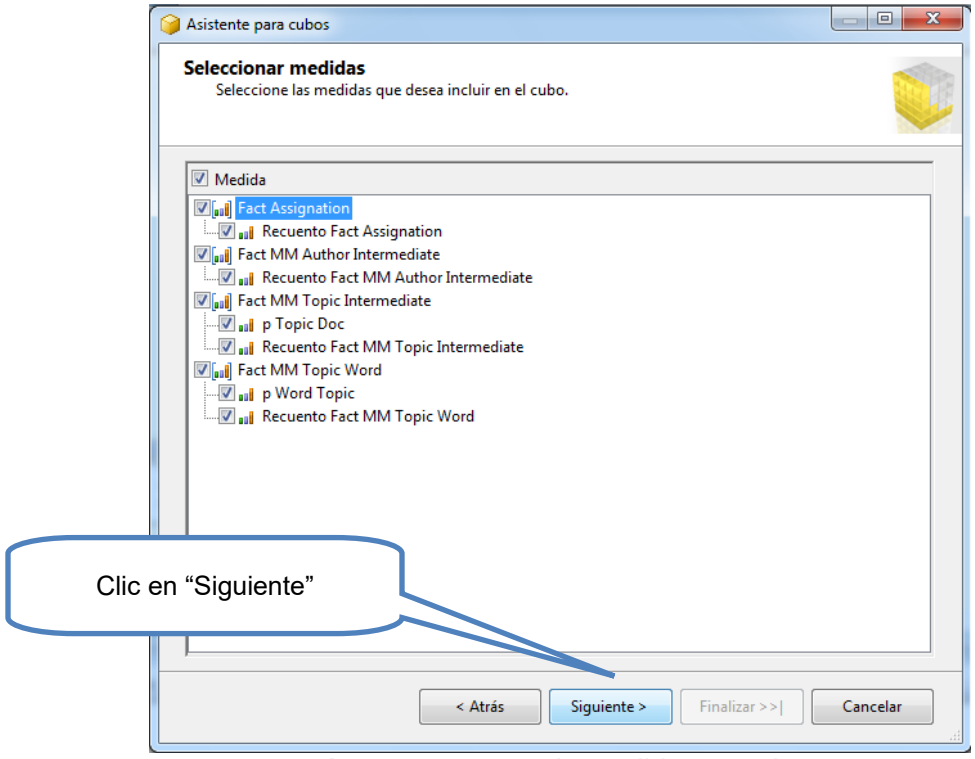


Figura 79. Grupo de medidas creadas

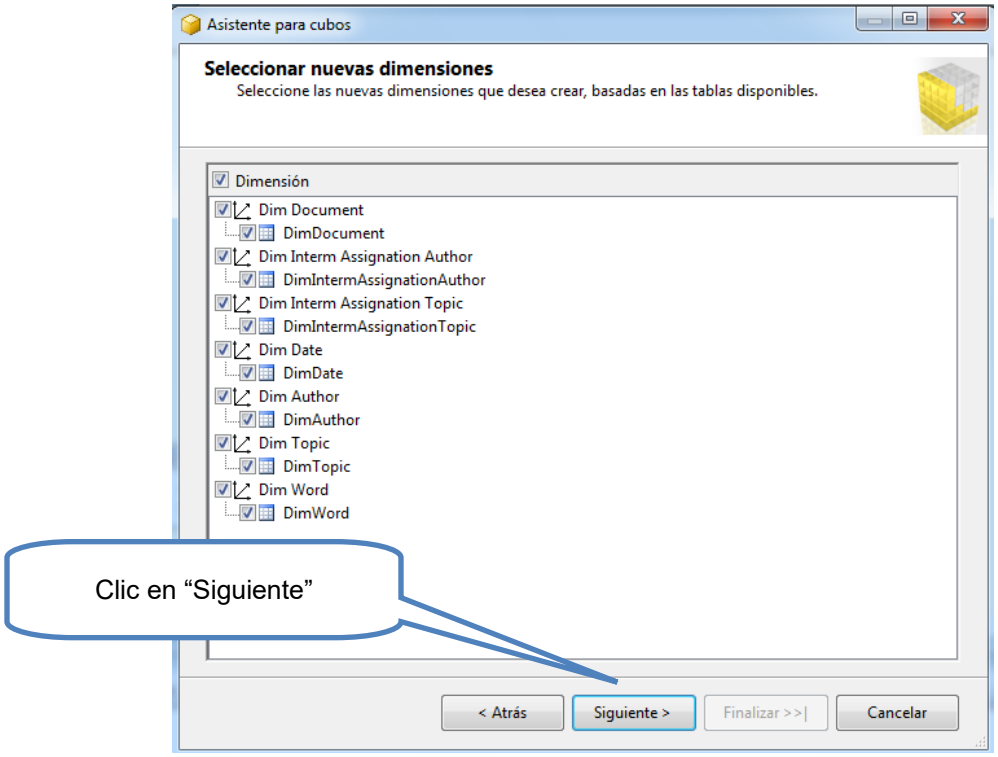


Figura 80. Dimensiones definidas para el cubo

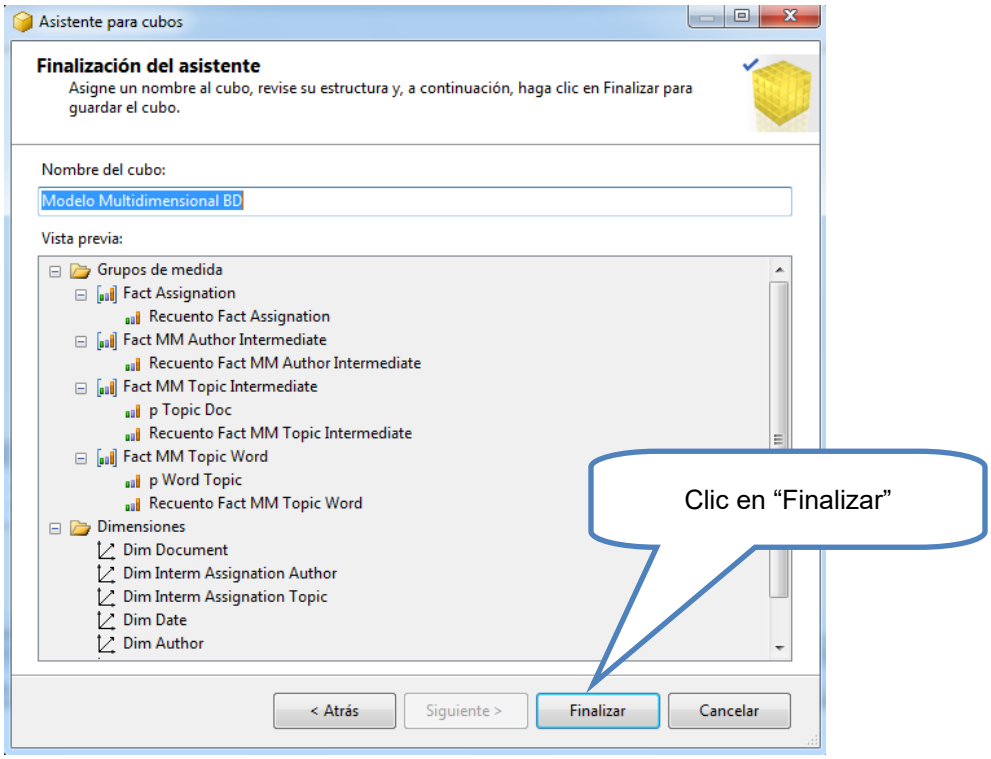


Figura 81. Finalización del asistente del nuevo cubo

Se obtiene el modelo lógico del modelo multidimensional propuesto (ver **Figura 82**).

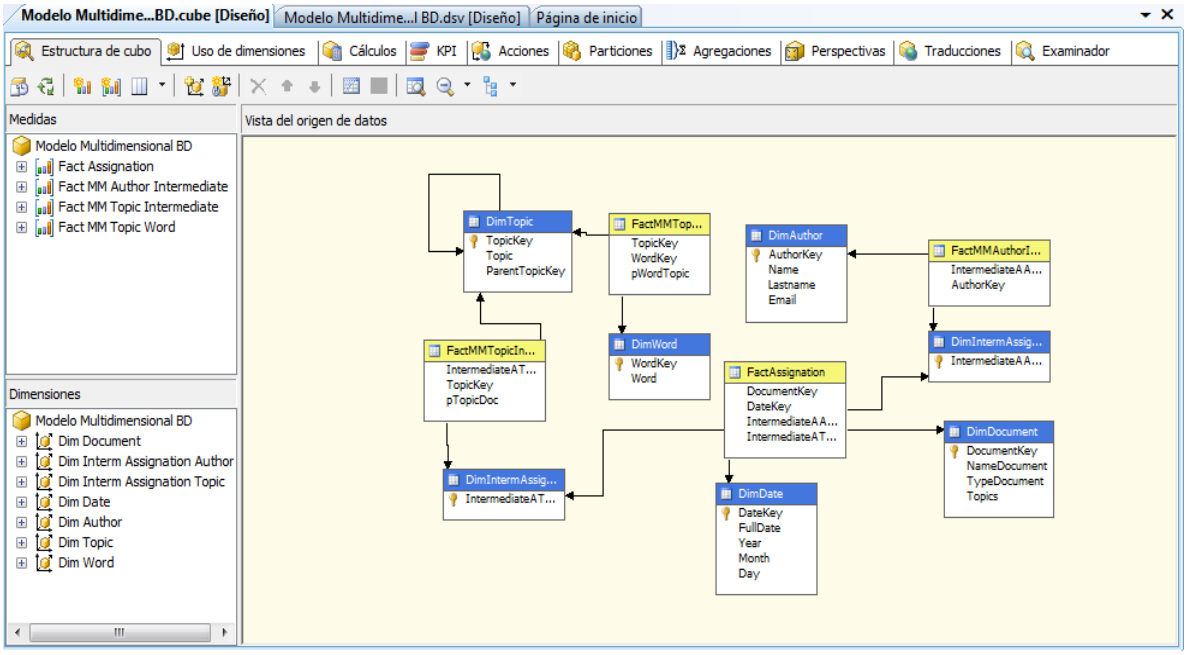


Figura 82. Modelo lógico

Séptimo: Se configuran las tablas de dimensión que tienen relación *muchos-a-muchos* (ver **Figura 83** y **Figura 84**). Por ejemplo se toma la tabla de dimensión *DimTopic*la cual tiene una relación *muchos-a-muchos* hacia la tabla de hecho *FactAssignation*.

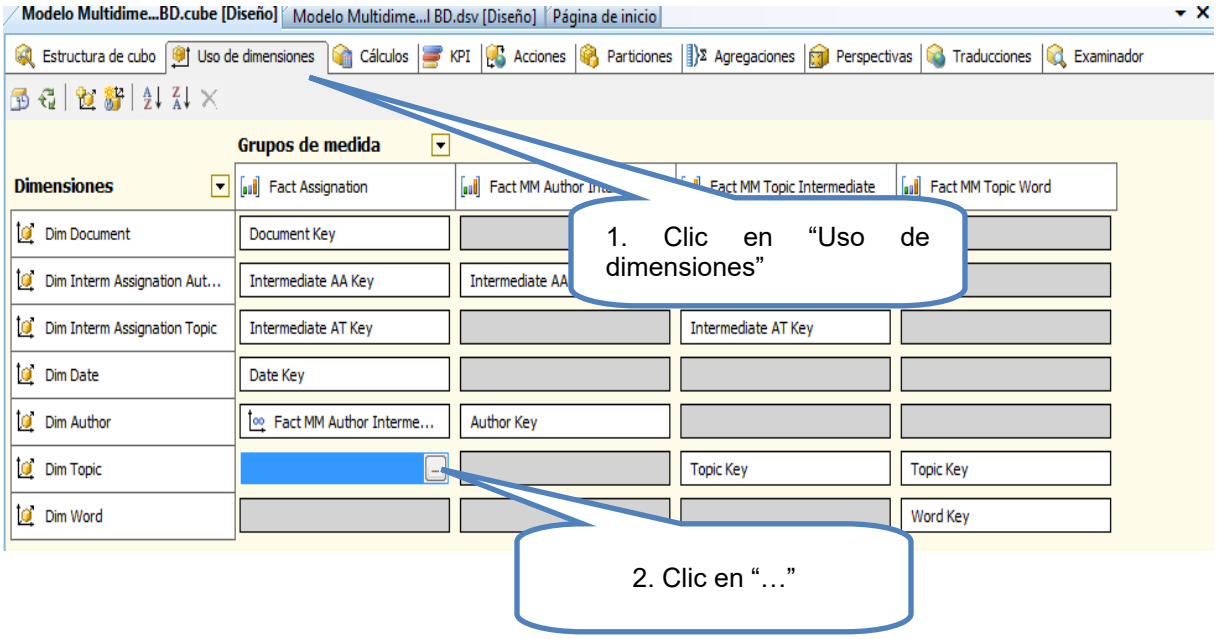


Figura 83. Configuración uso de dimensiones

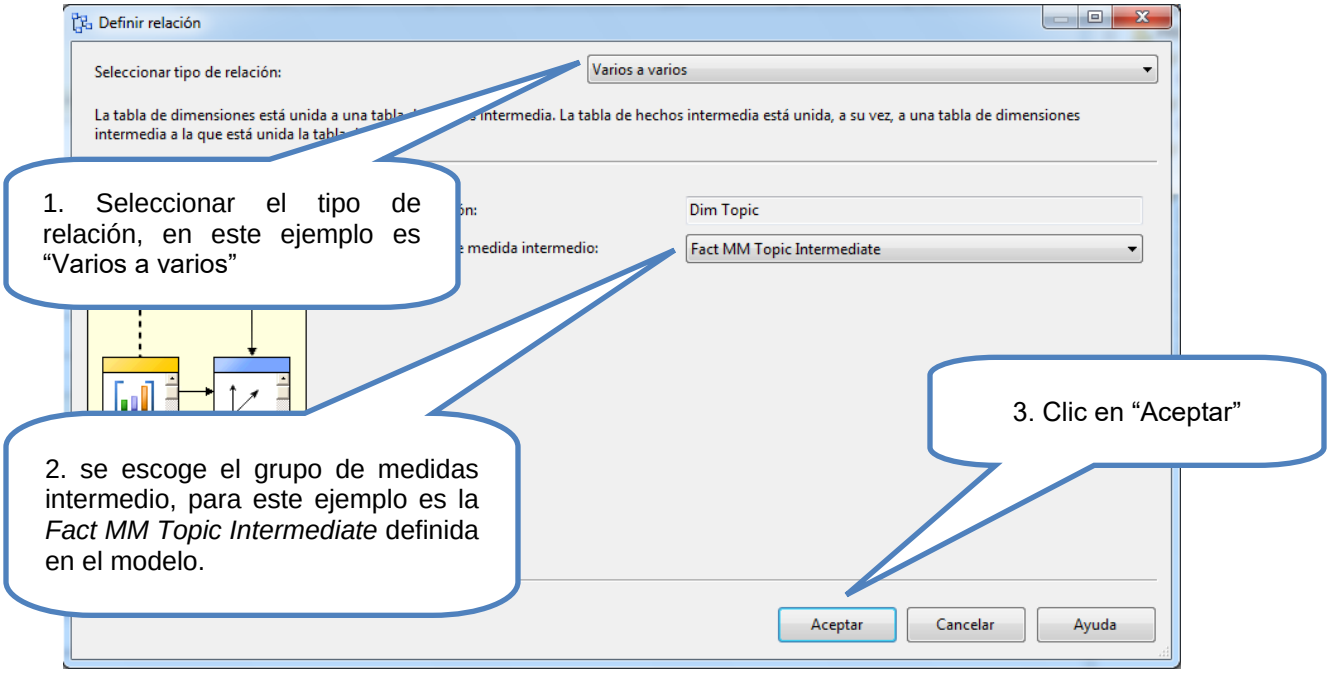


Figura 84. Seleccionar la relación Muchos-a-Muchos

De forma similar se procede para las dimensiones *DimAuthor, DimWord* y *DimDocument,* y obtener la siguiente configuración en el uso de dimensiones (ver Figura 85).

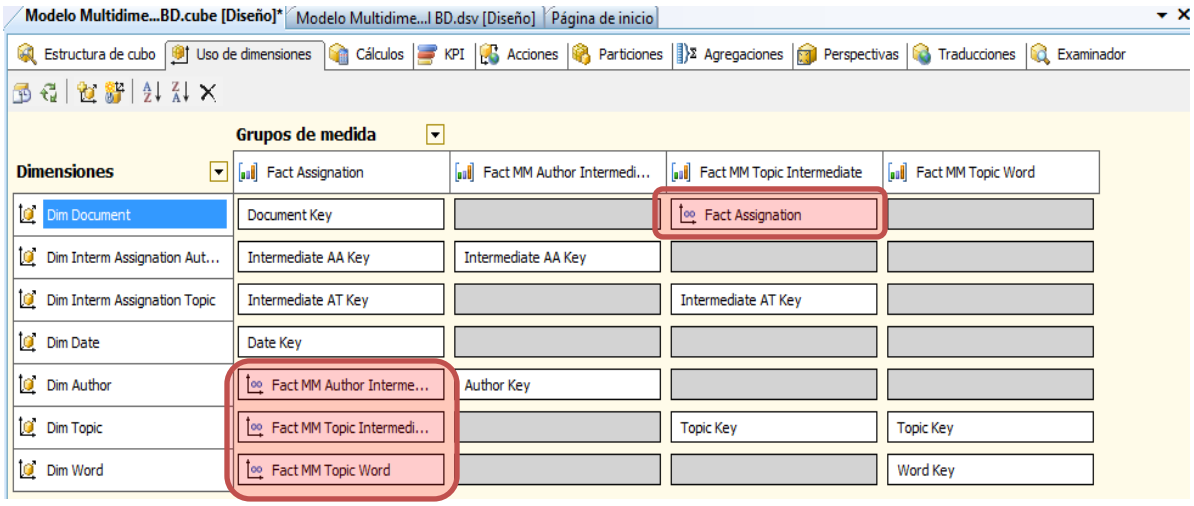


Figura 85. Configuración completa en el uso de dimensiones para DimDocument, DimAuthor, DimTopic y DimWord

Octavo: Se definen atributos y jerarquías en las dimensiones del modelo multidimensional en la estructura de cubo (ver **Figura 86**).

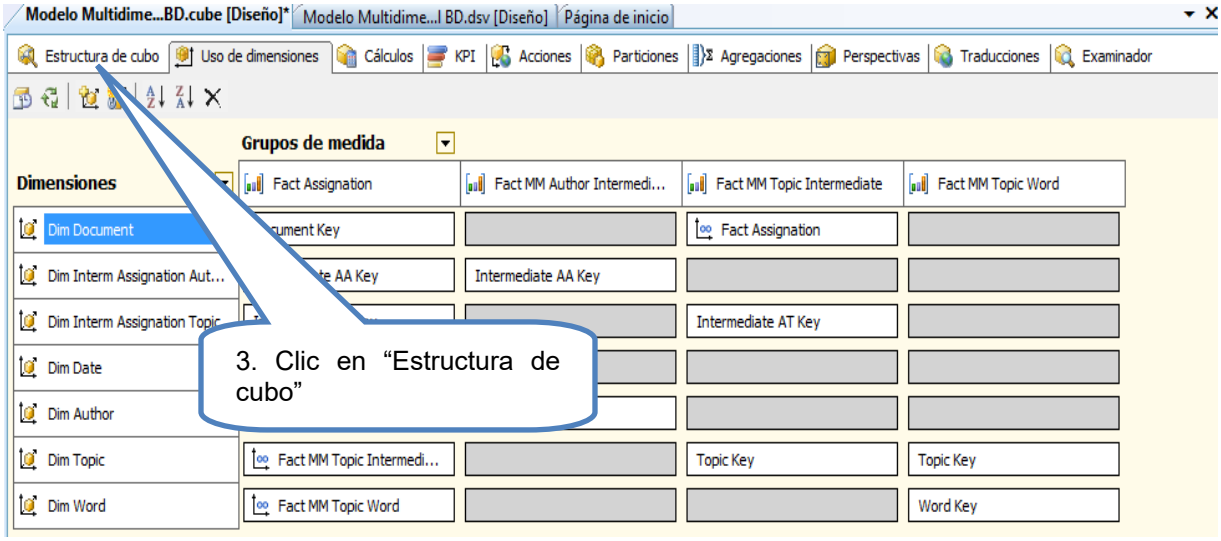


Figura 86. Seleccionar "Estructura de cubo"

Para adicionar atributos y definir las jerarquías es necesario tener en cuenta para cada dimensión su estructura y la forma en la cual se han jerarquizado sus atributos, por tanto se muestra como ejemplo la configuración para la dimensión *DimDocument*(ver **Figura 87**, **Figura 88** y **Figura 89**).

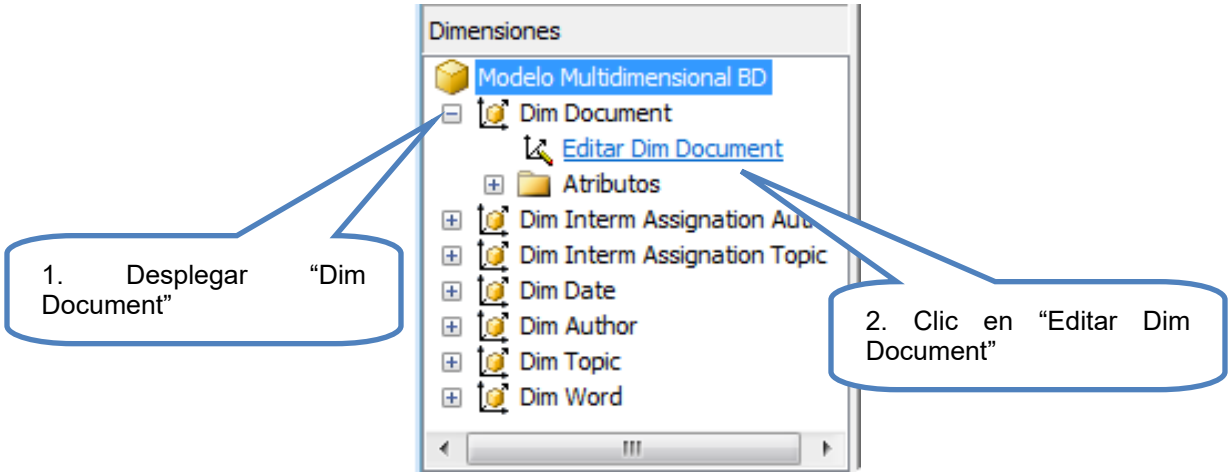


Figura 87. Editar DimDocument

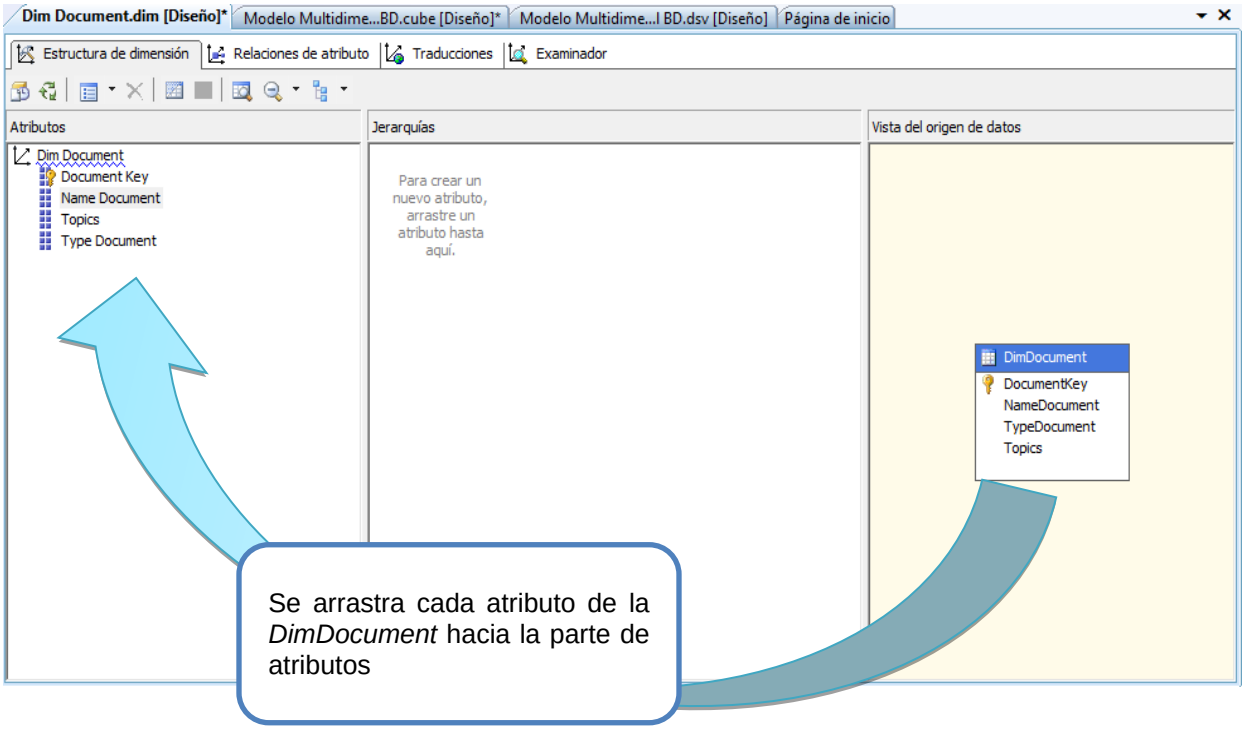


Figura 88. Adicionar atributos de la dimensión DimDocument

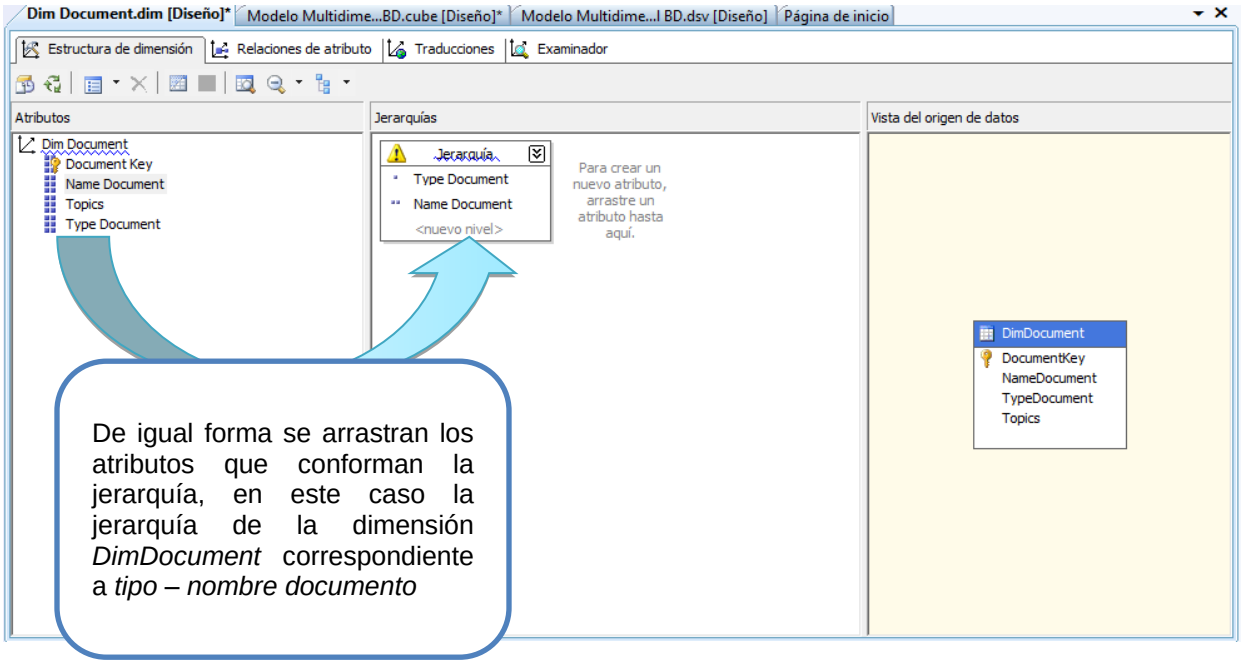


Figura 89. Definir la jerarquía para la dimensión DimDocument

De forma similar se definen los atributos y las jerarquías para las dimensiones *DimDate*, *DimAuthor* y *DimWord*. Para la dimensión *DimTopic* se debe tener un tratamiento especial.

Como la dimensión*DimTopic* se definió como una jerarquía padre-hijo[23] en la base de datos relacional, la herramienta de Analysis Services la detecta automáticamente, solo es necesario tener en cuenta la siguiente configuración.

- Configuración del padre (ver **Figura 90**, **Figura 91**, **Figura 92** y **Figura 93**)

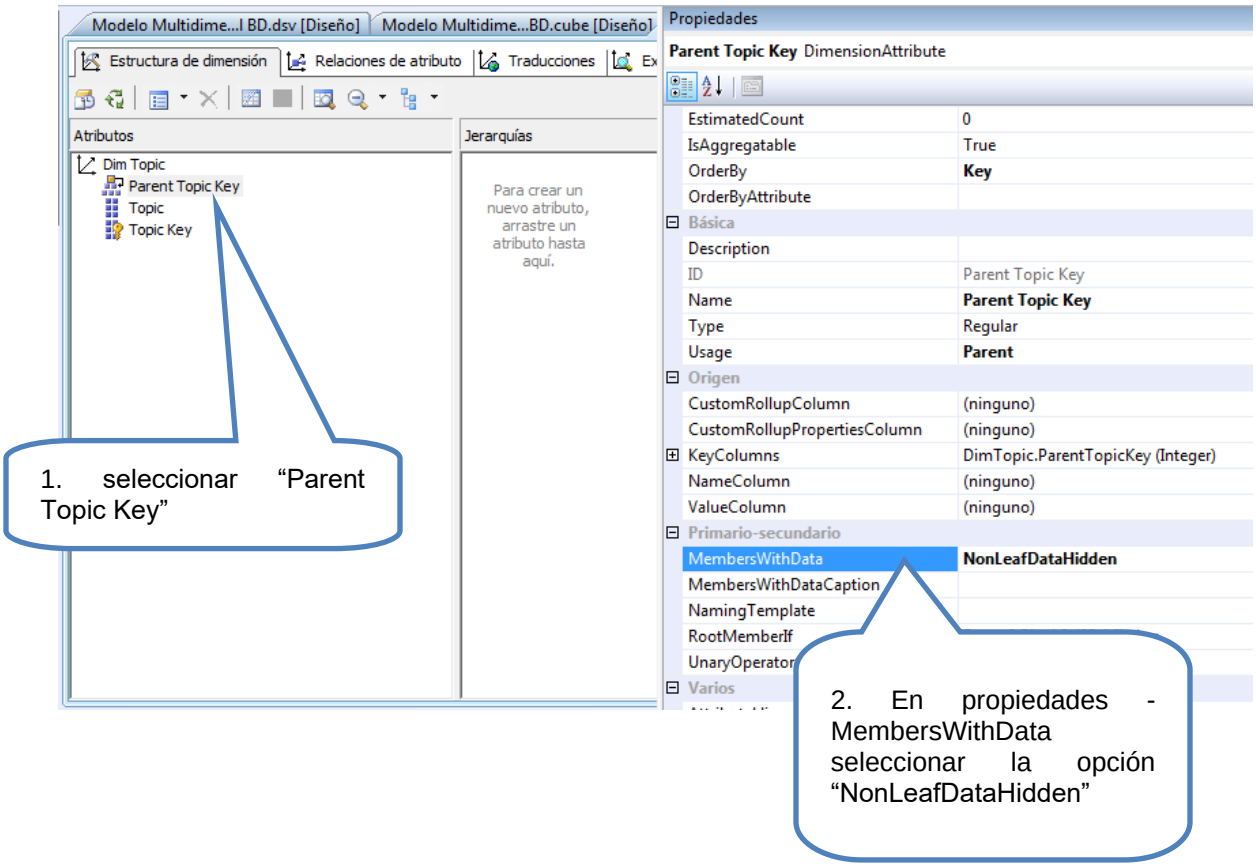


Figura 90. Configuración del padre en su propiedad *MembersWithData*

[23] El modelo jerárquico facilita relaciones 1:N (uno a muchos), de manera que un padre puede tener más de un hijo, todos ellos localizados en el mismo nivel, y un hijo solo puede tener un padre situado en el nivel inmediatamente superior al suyo.

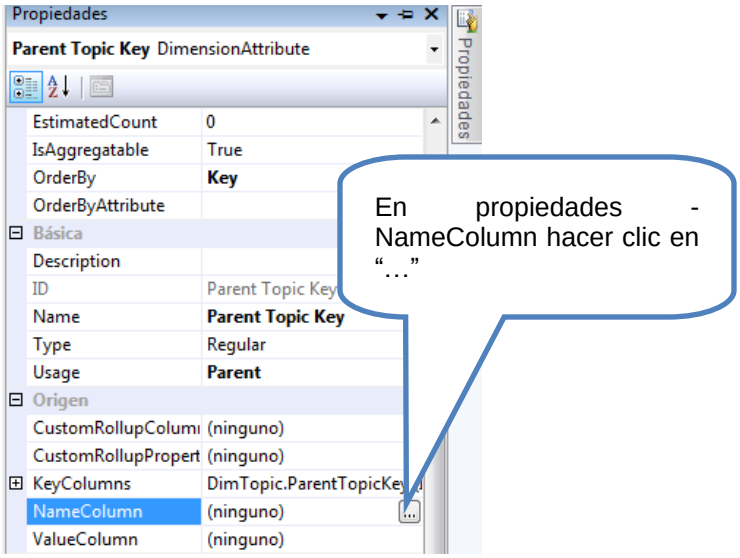


Figura 91. Configuración del padre en su propiedad *NameColumn*

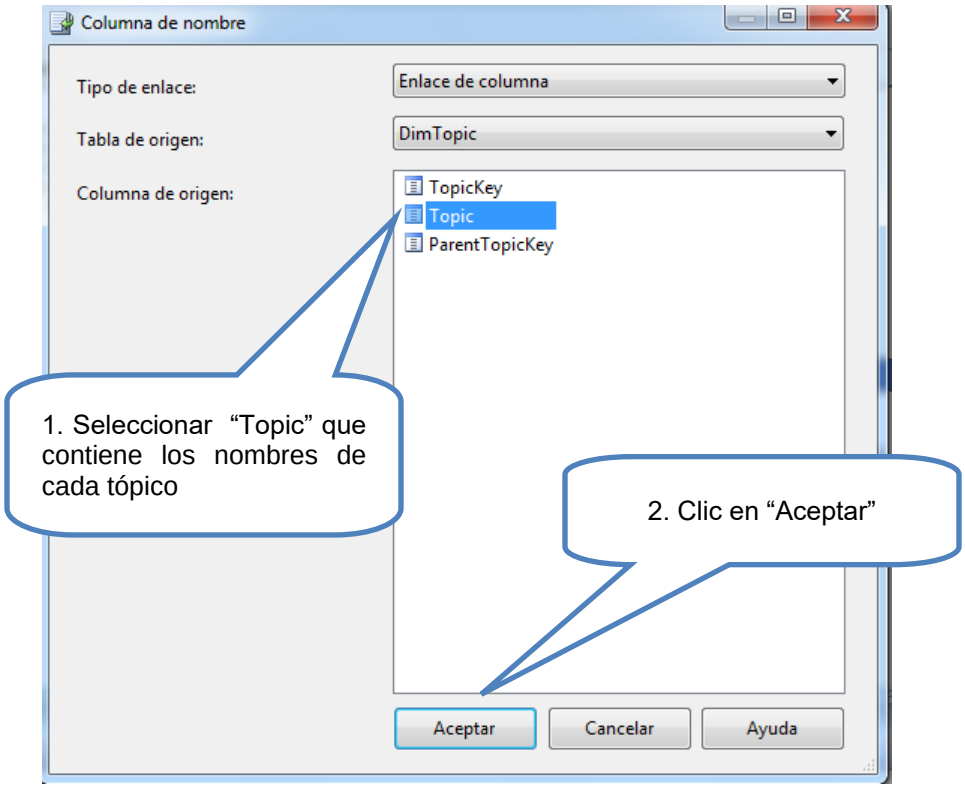


Figura 92. Seleccionar el nombre del tópico padre y finalizar

Quedando la siguiente configuración en las propiedades de *Parent Topic Key*

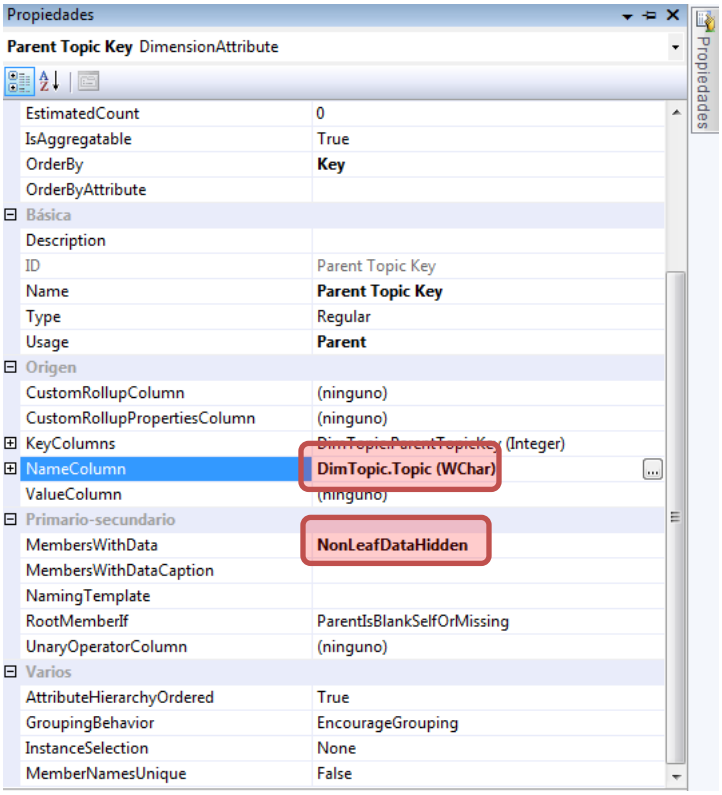


Figura 93. Configuración del padre

- Configuración del hijo (ver **Figura 94** y **Figura 95**)

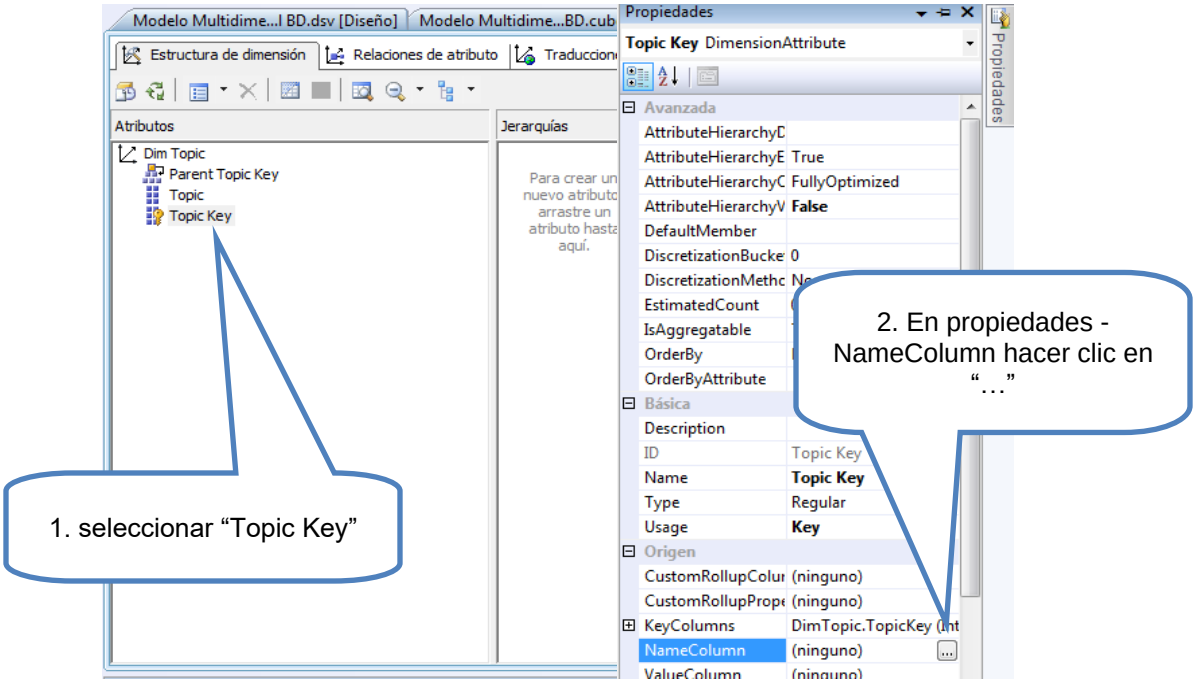


Figura 94. Configuración del hijo en su propiedad *NameColumn*

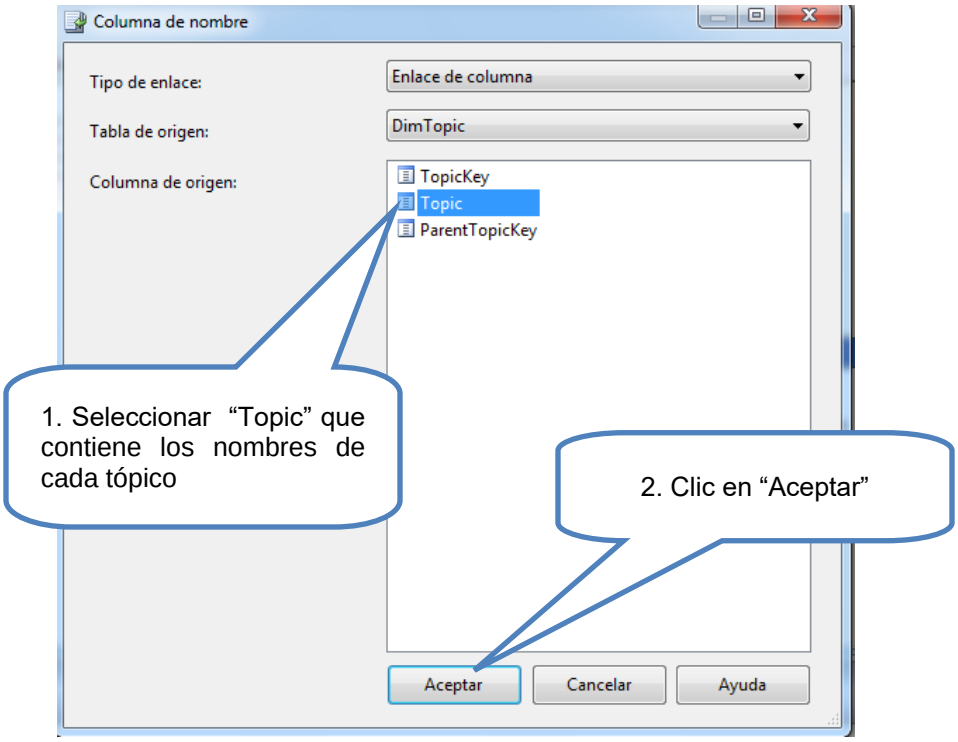


Figura 95. Seleccionar el nombre del tópico hijo y finalizar

Quedando la siguiente configuración en las propiedades de *Topic Key* (ver **Figura 96**).

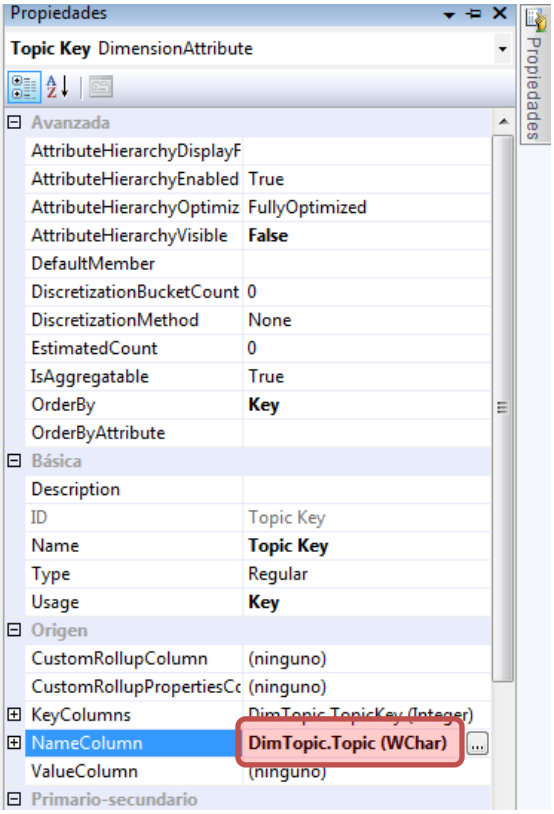


Figura 96. Configuración del hijo

ANEXO B – BASE DE DATOS RELACIONAL SQL SERVER 2008

Modelo Relacional (ver Figura 97)

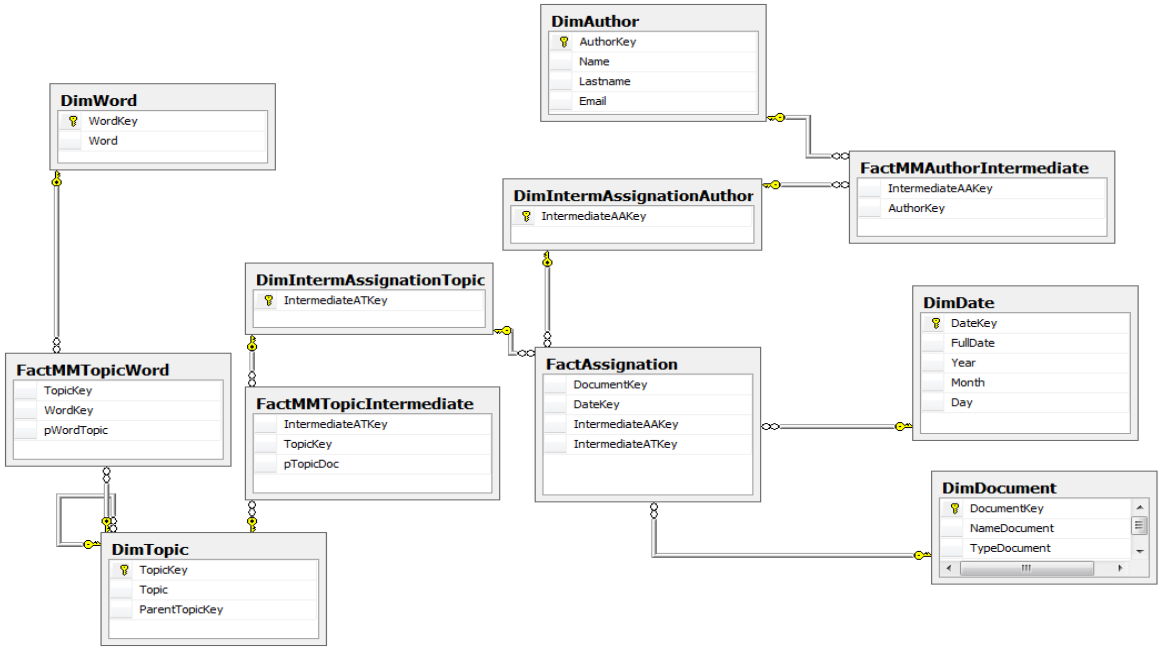


Figura 97. Modelo Relacional de la bodega de documentos

Estructura de las tablas:

DimWord

Atributo	Tipo de dato
WordKey	int
Word	Varchar(255)

DimAuthor

Atributo	Tipo de dato
AuthorKey	int
Name	Varchar(255)
Lastname	Varchar(255)
Email	Varchar(255)

DimDate

Atributo	Tipo de dato
DateKey	int
FullDate	Varchar(255)
Year	Int
Month	Int
Day	int

DimDocument

Atributo	Tipo de dato
DocumentKey	int
NameDocument	Varchar(255)
TypeDocument	Varchar(255)
Topics	Varchar(255)

DimTopic

Atributo	Tipo de dato
TopicKey	int
Topic	Varchar(255)
ParentTopicKey	int

FactMMTopicIntermeadite

Atributo	Tipo de dato
IntermediateATKey	int
TopicKey	int
pTopicDoc	Numeric(23,22)

FactMMTopicWord

Atributo	Tipo de dato
TopicKey	int
WordKey	int
pWordTopic	Numeric(11,10)

DimDimIntermAssignationTopic

Atributo	Tipo de dato
IntermediateATKey	int

FactAssignation

Atributo	Tipo de dato
DocumentKey	int
DateKey	int
IntermediateAAKey	int
IntermediateATKey	int

DimIntermAssignationAuthor

Atributo	Tipo de dato
IntermediateAAKey	int

FactMMAuthorIntermediate

Atributo	Tipo de dato
IntermediateAAKey	int
AuthorKey	int

ANEXO C – CÓDIGO MDX TEXTMEASURE_TOPICS_PROBAB

Medida TextMeasure_Topics_Probab

La dimensión *DimDocument* tiene un atributo *Topics*, el cual contiene la información correspondiente a los tópicos o temas que trata el documento. Teniendo en cuenta esto, la medida textual *Textmeasure_Topics_probab* se crea de la siguiente manera.

En la estructura de cubo se crea una medida numérica (ver **Figura 98**, **Figura 99** y **Figura 100**) en el grupo de medidas de la *Fact Assignation* que asocia la llave primaria de la tabla de dimensión *DimDocument*.

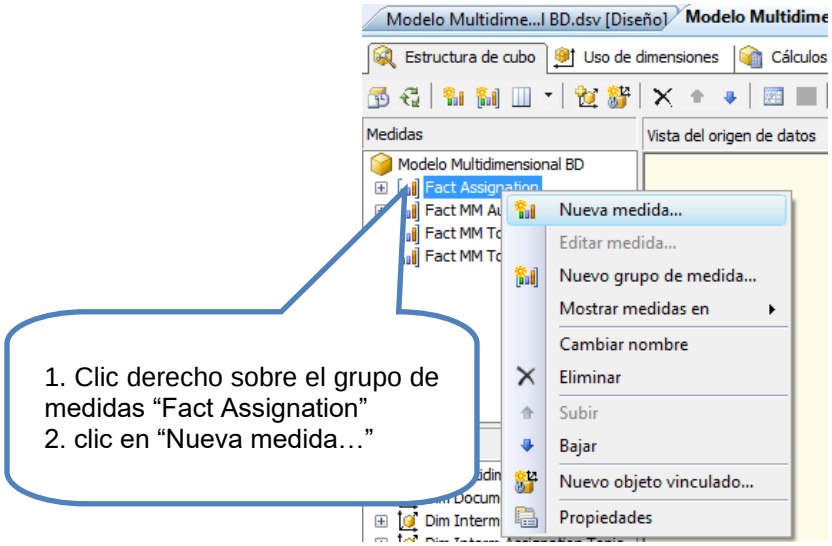


Figura 98. Nueva medida numérica en el grupo de medida *FactAssignation*

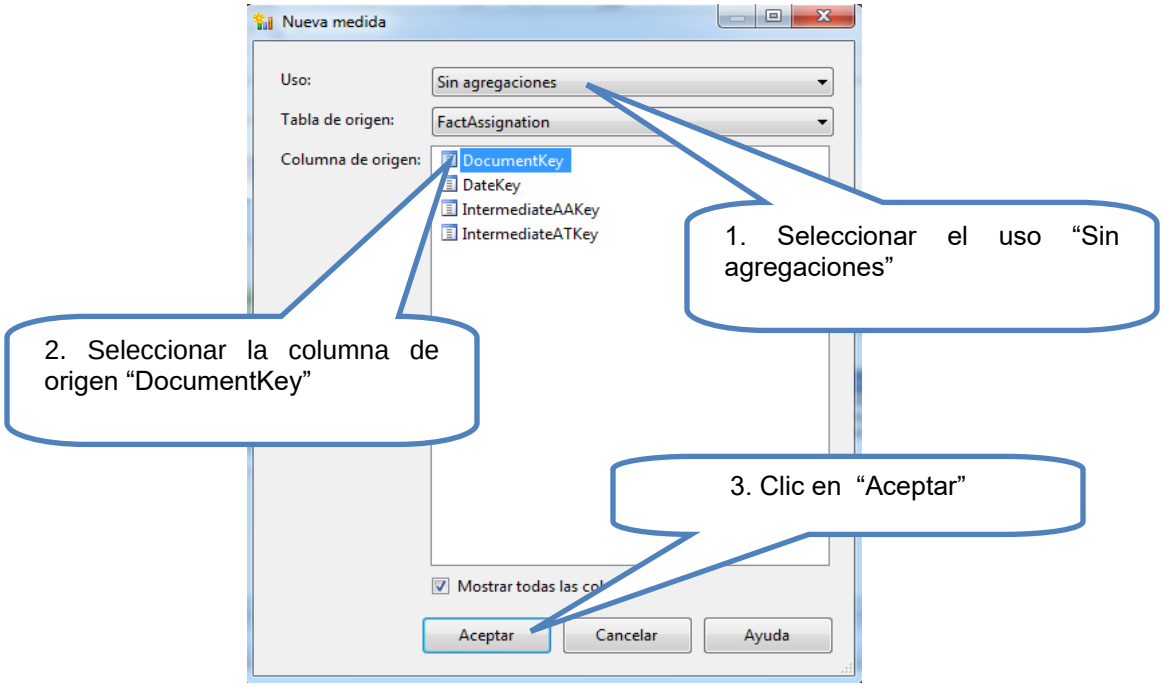


Figura 99. Seleccionar la nueva medida numérica con *DocumentKey*

Quedando adicionado la medida numérica DocumentKey y la siguiente configuración en las propiedades.

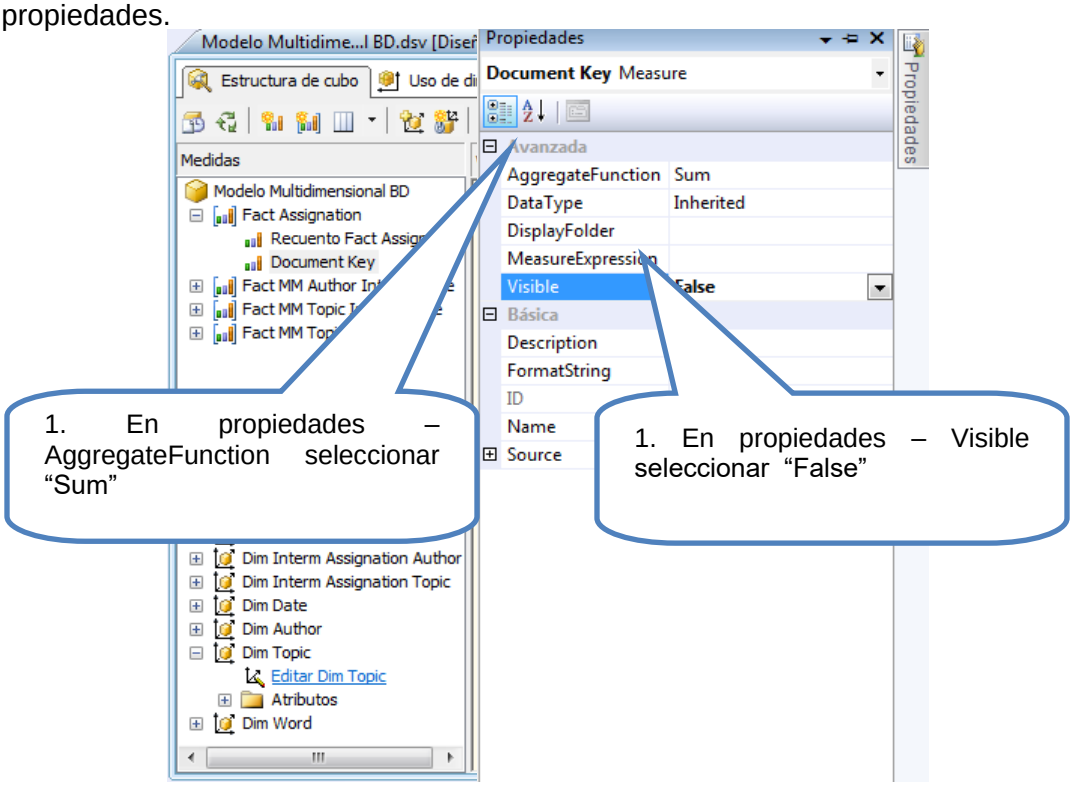


Figura 100. Configuración de la medida numérica *DocumentKey* en sus propiedades

Con base en la medida numérica anterior, se crea una medida textual calculada con MDX en la pestaña *Cálculos* de AnalysisServices (ver **Figura 101** y **Figura 102**), que extrae todos los tópicos con sus probabilidades del atributo *Topics* de la dimensión *DimDocument*.

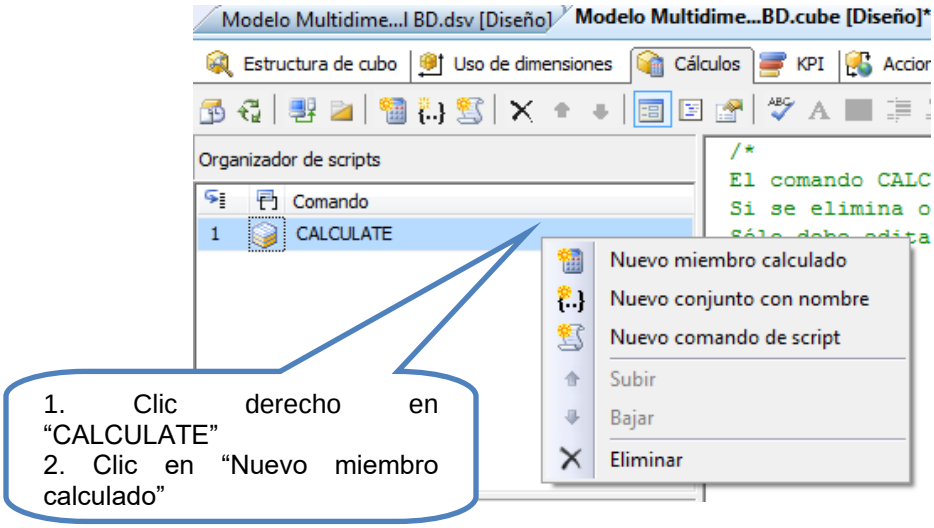


Figura 101. Nueva medida calculada para [Annotations-Topics]

A continuación se presenta el código MDX correspondiente a la medida calculada [Annotations-Topics]:

```
IIF([Measures].[Document Key] = 0, NULL, StrToMember("[Dim Document].&["+
VBA!CStr([Measures].[Document Key]) + "]").PROPERTIES("Topics"))
```

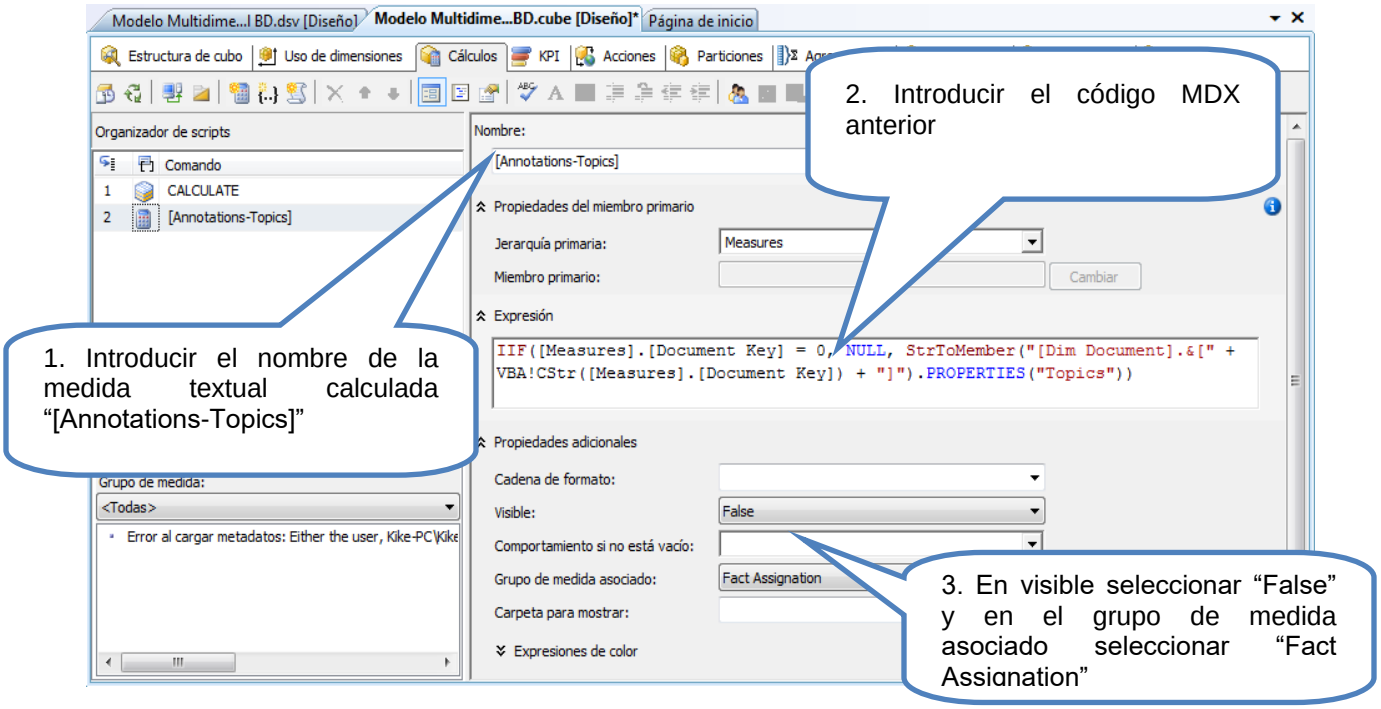


Figura 102. Creación de la medida calculada [Annotations-Topics] con MDX

Al tener de la anterior medida textual calculada los tópicos con sus probabilidades, se crea una medida textual calculada (ver **Figura 103** y Figura 104) que asocia las dimensiones del modelo multidimensional propuesto, para mostrar a través de las consultas, los tópicos con sus probabilidades de acuerdo a las dimensiones relacionadas, adicionalmente permitir que la función de agregación creada en el procedimiento almacenado, promedie los tópicos que se repiten cuando se realizan las tradicionales operaciones OLAP[24] (drill-down, roll-up).

Código MDX [Topics-Prob]:

```
ManejodeTexto.ManejodeTexto.Cadena.CadenaProbTopics(
        GENERATE(
                EXISTING {([Dim Document].[Document Key].[Document Key],
                [Measures].[Annotations-Topics])},
                IIf([Measures].[Annotations-Topics]=NULL, NULL,
                [Measures].[Annotations-Topics]+"; ")
        )
)
```

Se crea la medida textual calcula en Analysis Services:

[24] Por sus siglas en inglés, On Line Analytical Processing

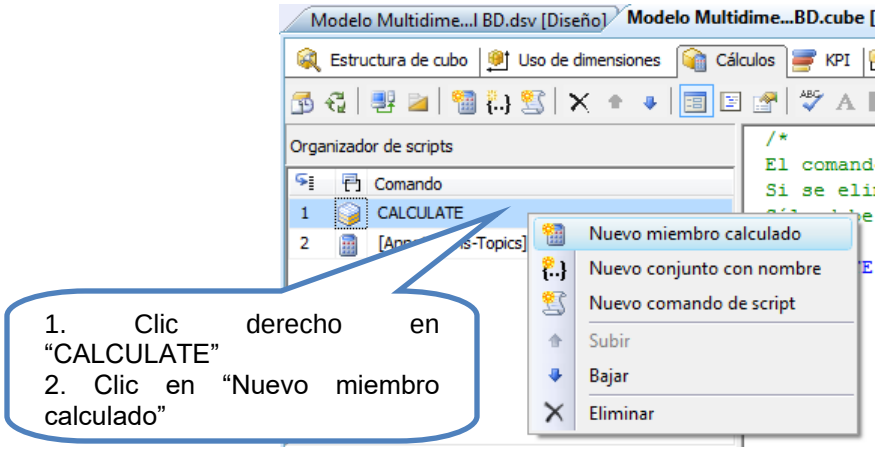


Figura 103. Nueva medida calculada para [Topics-Prob]

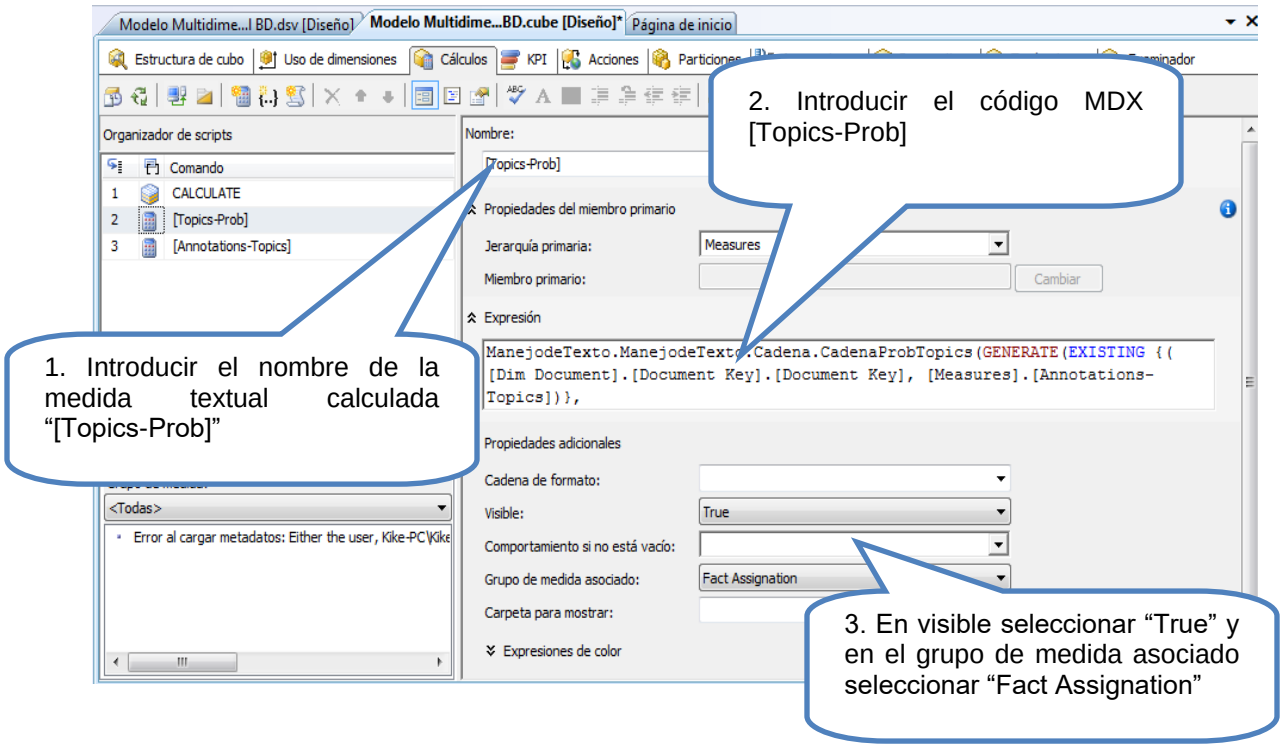


Figura 104. Creación de la medida calculada [Topics-Prob] con MDX

ANEXO D - CÓDIGO FUENTE DEL PROCEDIMIENTO ALMACENADO Y CONFIGURACIÓN EN ANALYSIS SERVICES

Código Assembly (Función de Agregación)

```
namespace ManejodeTexto
{
publicclassCadena
   {
ArrayList ArrayTopics = newArrayList();
ArrayList ArrayTopicsProba = newArrayList();
string CadenaFinal = "";

publicstring CadenaProbTopics(string TopicsAndProbalities)
      {
if (TopicsAndProbalities != "")
         {
string[] ArrayTopicsProbabilities = TopicsAndProbalities.Split(';');
int j = 0;
for (int i = 0; i < ArrayTopicsProbabilities.Length - 1; i++)
            {
string[] prob = ArrayTopicsProbabilities[i].Split(':');
if (prob.Length == 2)
               {
                  ArrayTopics.Add(prob[0].Trim());
                  ArrayTopicsProba.Add(prob[1].Trim());
               }
            }
int cont = 0;
for (int i = 0; i < ArrayTopics.Count; i++)
            {
if (ArrayTopics[i] != "")
               {
string tempTopic = ArrayTopics[i].ToString();
double tempProb = Convert.ToDouble(ArrayTopicsProba[i].ToString());

this.Clear(i, ArrayTopics);
this.Clear(i, ArrayTopicsProba);
ArrayList position = FindCadena(tempTopic);
if (position.Count > 0)
                  {
foreach (int item in position)
                     {
                        tempProb += Convert.ToDouble(ArrayTopicsProba[item]);

                     }
foreach (int item in position)
                     {
this.Clear(item, ArrayTopics);
this.Clear(item, ArrayTopicsProba);
                     }
int num = position.Count + 1;
                     tempProb = (tempProb / num) * 100;

                     CadenaFinal += tempTopic + ": " + String.Format("{0:0.00}", tempProb) + "%; ";//

                  }
else
                  {
double tmp = Convert.ToDouble(tempProb) * 100;
                     CadenaFinal += tempTopic + ": " + String.Format("{0:0.00}", tmp.ToString()) + "%; ";
                  }
               }
            }
return CadenaFinal;
         }
else
         {
returnnull;
         }
      }
```

```
privateArrayList FindCadena(string cadena)
      {
ArrayList position = newArrayList();
for (int i = 0; i < ArrayTopics.Count; i++)
         {
if (ArrayTopics[i].Equals(cadena))
            {
               position.Add(i);
            }
         }
return position;
      }
privatevoid Clear(int item, ArrayList vector)
{
         vector[item] = "";
      }
   }
}
```

Este procedimiento almacenado permite a las medidas textuales realizar las tradicionales operaciones OLAP(drill-down, roll-up) a traves de su funcion de agregacion, para el caso de estudio, promediar las probabilidades de los topicos que se repiten y darle un formato de salida.Por tanto,al tener compilado el anterior codigo fuente, el proyecto genera una .DLL (Manejodetexto.dll) la cual debe ser configurada (ver **Figura 105**, **Figura 106** y **Figura 107**) de la siguiente manera en el explorador de soluciones del proyecto de analysis services.

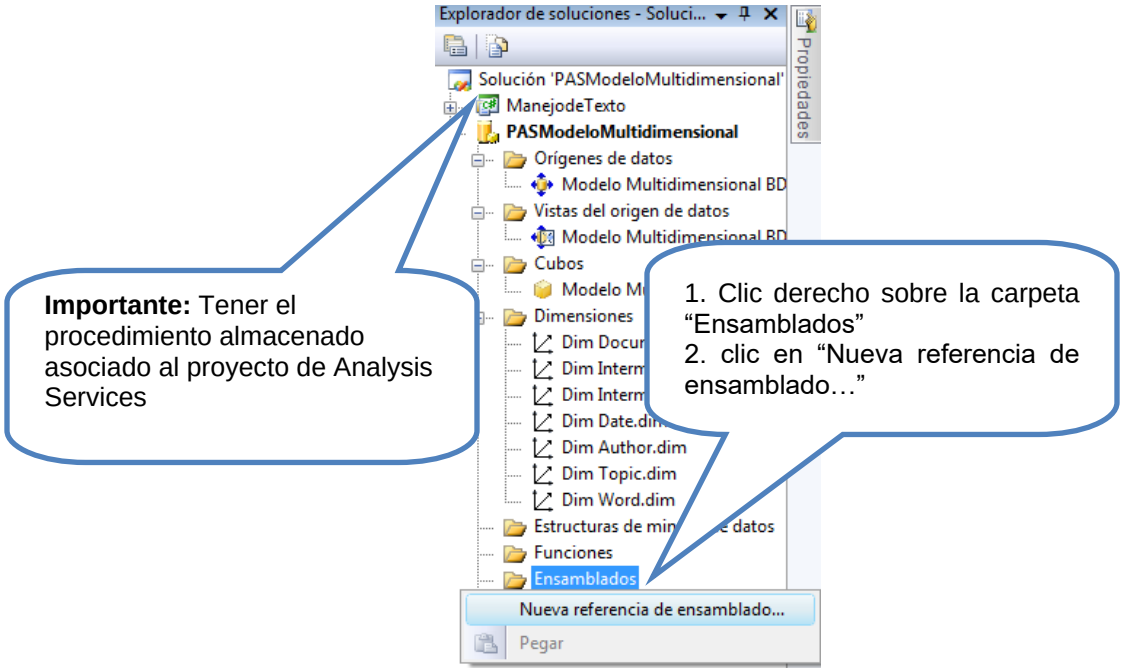


Figura 105. Definir nueva referencia al procedimiento almacenado

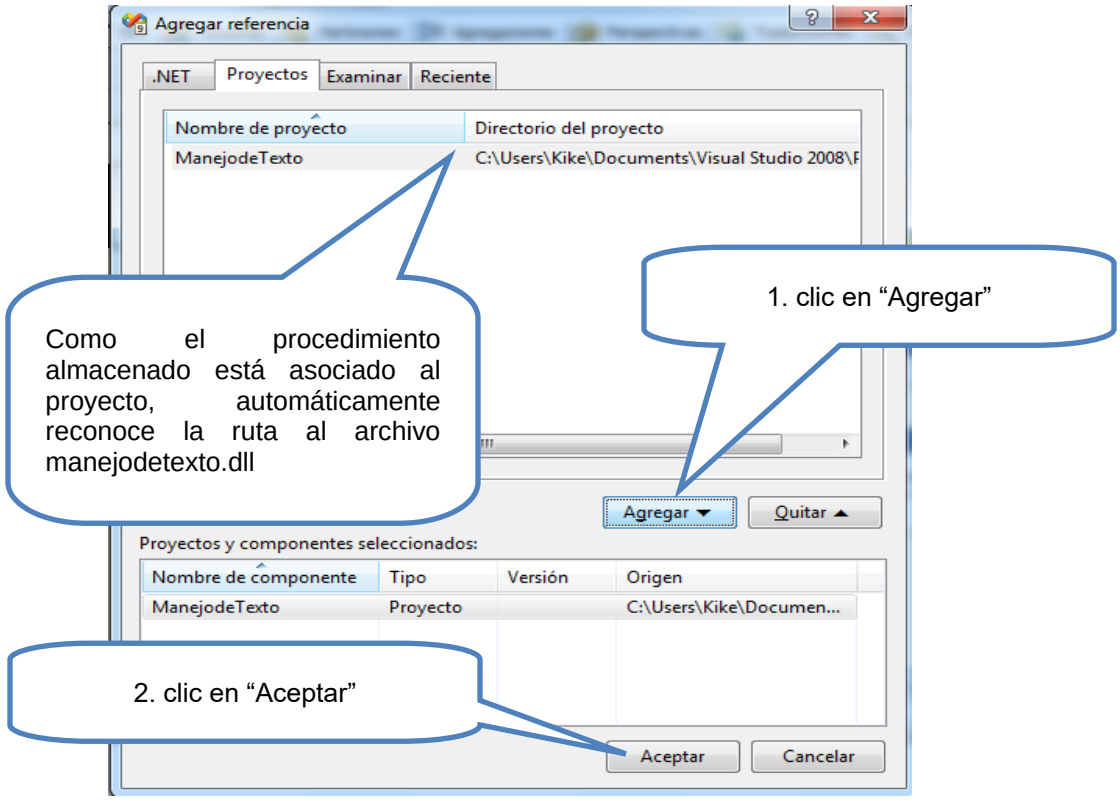


Figura 106. Agregar el archivo manejotexto.dll al proyecto de Analysis Services

Quedando en el explorador de soluciones.

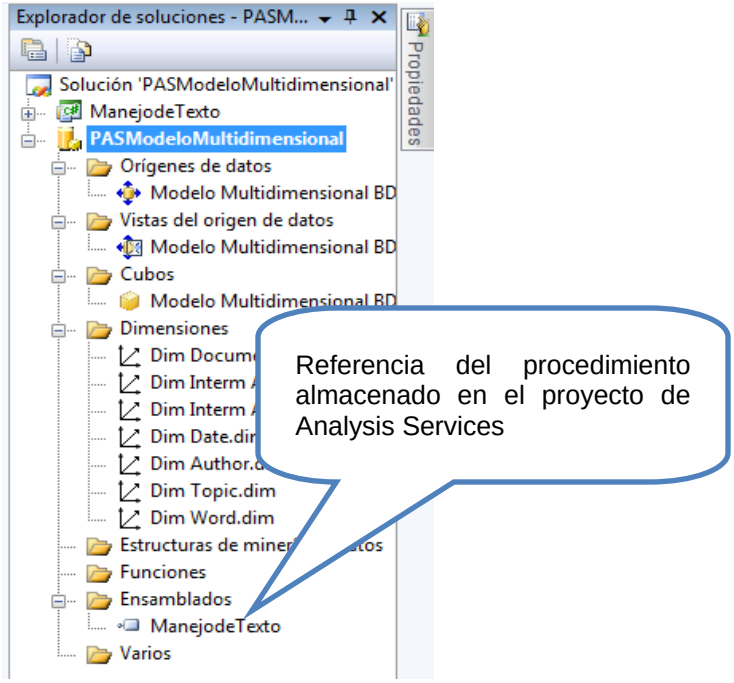


Figura 107. Procedimiento almacenado adicionado al proyecto de Analysis Services

ANEXO E – CÓDIGO MDX TEXTMEASURE_WORD_PROBAB

Medida TextMeasure_Word_Probab

Se crea una medida textual calculada (ver **Figura 108** y **Figura 109**) que permite obtener las palabras o términos con sus probabilidades de acuerdo a los tópicos definidos en la dimensión DimTopic.

Código MDX [Words-Prob]:

```
GENERATE({[Dim Word].[Word Key].[Word Key]},IIF([Measures].[Recuento Fact MM Topic
Word]=0, NULL,([Dim Word].[Word Key].PROPERTIES("Word")+": " + CSTR(([Measures].[p Word
Topic].VALUE / ([Dim Topic].[Topics].CURRENTMEMBER, [Measures].[Recuento Fact MM Topic
Word]))*100) + "%; ")))
```

Se crea la medida textual calcula en Analysis Services:

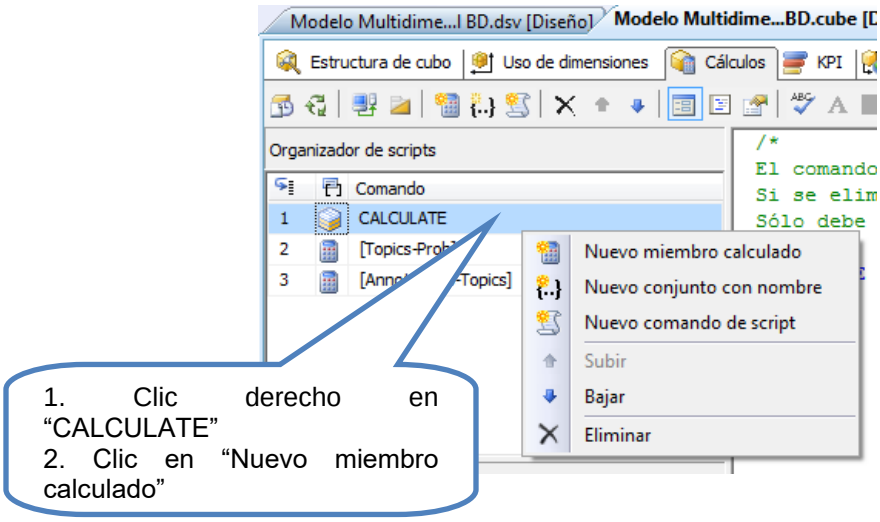


Figura 108. Nueva medida calculada para [Words-Prob]

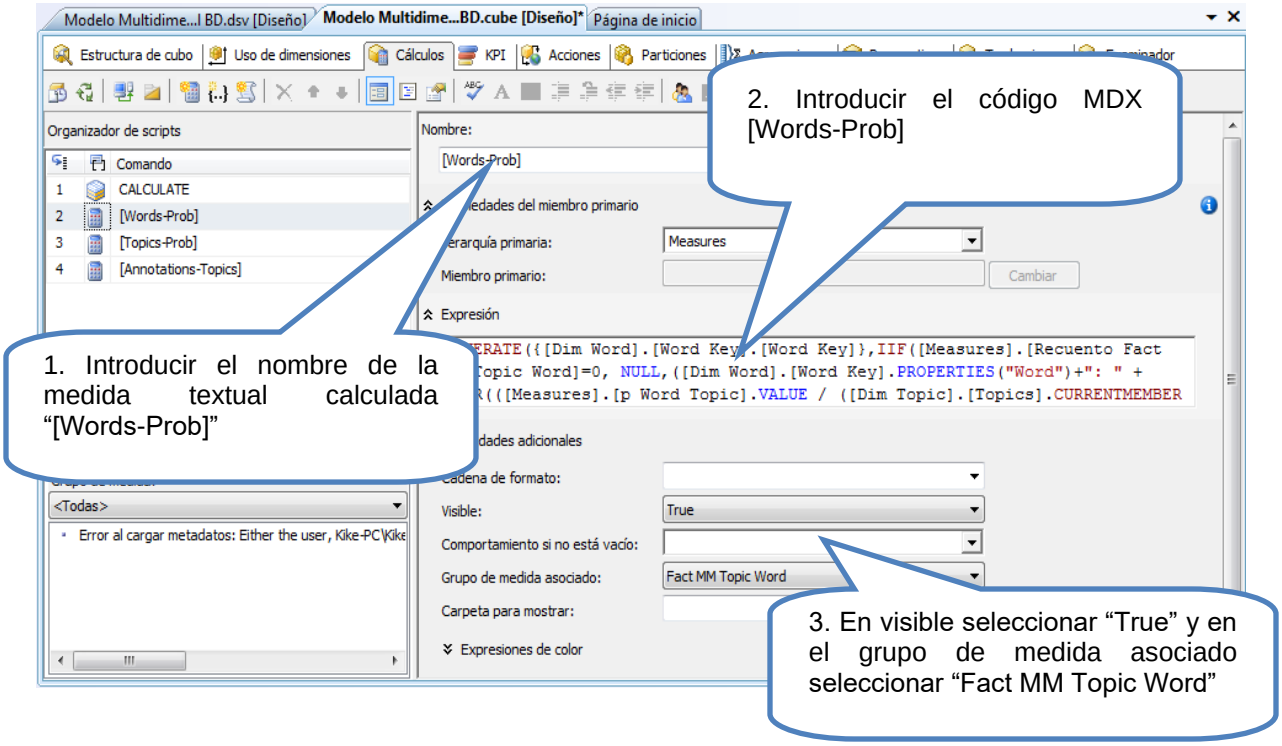


Figura 109. Creación de la medida calculada [Words-Prob] con MDX

ANEXO F – CÓDIGO MDX TEXTMEASURE_DOCUMENTS

Medida TextMeasure_Documents

Para esta medida textual se necesita crear una medida textual calculada que referencie de la dimensión *DimDocument* el atributo *Name Document*, el cual contiene la información correspondiente al nombre del documento. Teniendo en cuenta lo anterior y la medida numérica *DocumentKey*, la medida textual *Textmeasure_Documents* se crea de la siguiente manera (ver **Figura 110**, **Figura 111**, **Figura 112** y Figura 113).

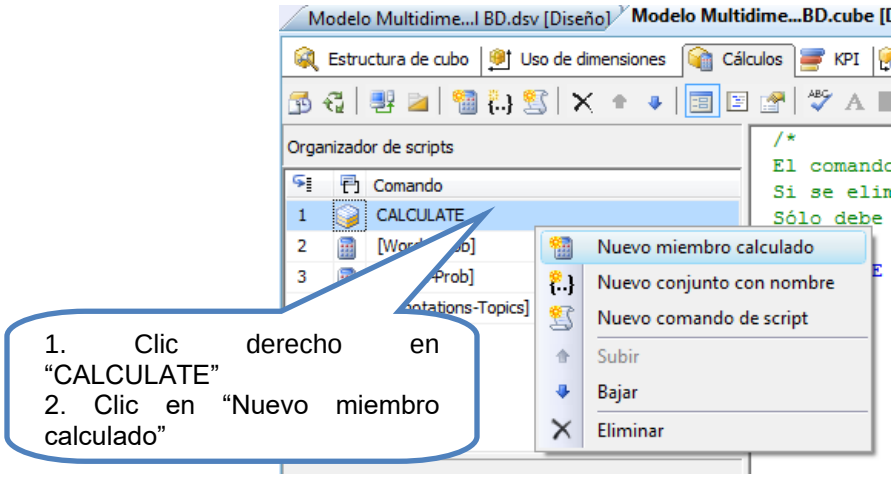


Figura 110. Nueva medida calculada para [Annotations-Docs]

A continuación se presenta el código MDX correspondiente a la medida calculada [Annotations-Docs]:

```
IIF([Measures].[Document Key] = 0, NULL, StrToMember("[Dim Document].&["+
VBA!CStr([Measures].[Document Key]) + "]").PROPERTIES("Name Document"))
```

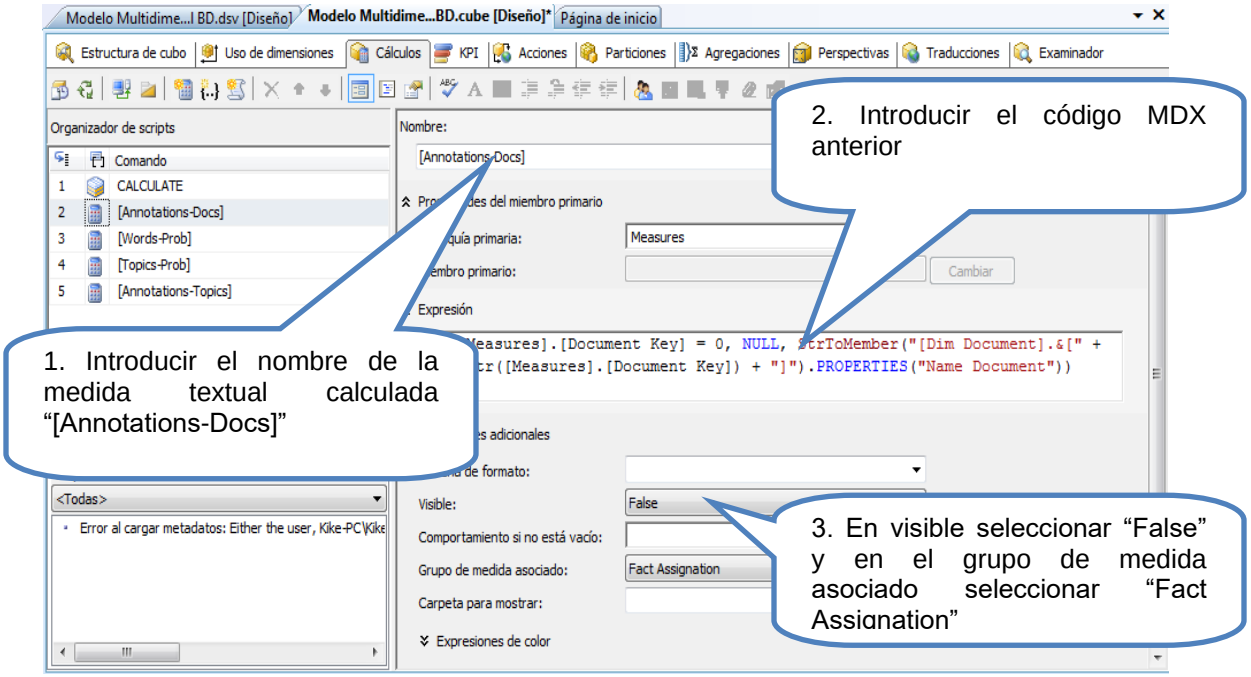


Figura 111. Creación de la medida calculada [Annotations-Docs] con MDX

Al tener los documentos de la medida textual anterior, se crea una medida textual calculada que permite asociar las dimensiones del modelo multidimensional propuesto, para mostrar a través de las consultas, los documentos con sus probabilidades de acuerdo a las dimensiones relacionadas, adicionalmente permitir que la función de agregación, implícita en el código MDX, promedie los documentos que se repiten cuando se realizan las tradicionales operaciones OLAP (drill-down, roll-up).

Código MDX [Documents]:

```
GENERATE(EXISTING {([Dim Document].[Document Key].[Document Key],
[Measures].[Annotations-Docs])},
IIf([Measures].[Annotations-Docs]=NULL, NULL, [Measures].[Annotations-Docs]+":
"+CSTR(([Measures].[p Topic Doc].VALUE/ ([Dim Topic].[Topics].CURRENTMEMBER,
[Measures].[Recuento Fact MM Topic Intermediate]))* 100)+"%; "))
```

Creación de la medida textual calcula en Analysis Services:

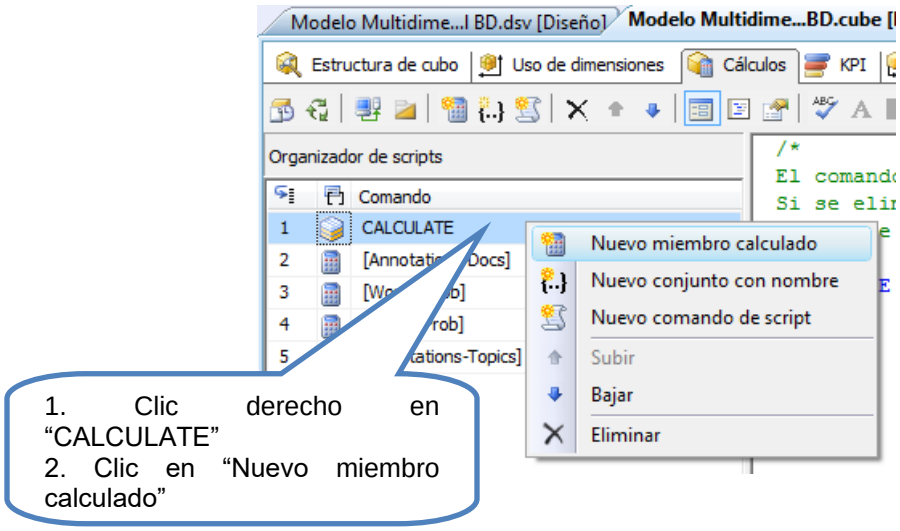


Figura 112. Nueva medida calculada para [Documents]

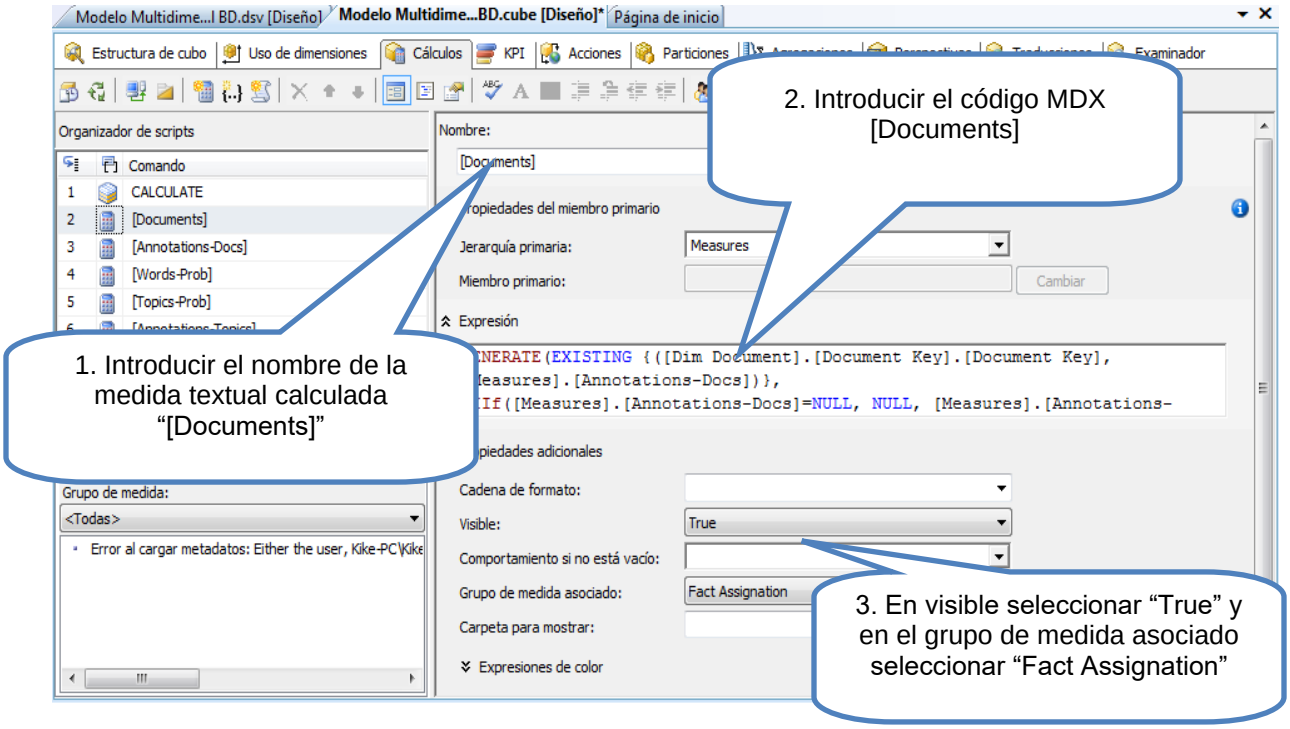


Figura 113. Creación de la medida calculada [Documents] con MDX

ANEXO G – APLICACIÓN PLSA

El conjunto de probabilidades correspondientes a las dos medidas probabilísticas P(z|d) (*pTopicDoc) y* P(w|z) (*pWordTopic)* definidas en el modelo multidimensional, se obtienen del algoritmo PLSA[25] implementado por HANG CHEN de la Universidad de Illinois, que fue desarrollada en lenguaje C++. Para utilizar el algoritmo PLSA se realizó una pequeña aplicación que permite hallar el conjunto de probabilidades para los tópicos del último nivel. A continuación se define los pasos para obtener las probabilidades en mención:

- **Paso 1:** Se copia el conjunto de archivos "*matrizTDFIDF_grupo_X_nivel_X.txt*" generados por el algoritmo IGBHSK modificado correspondientes al penúltimo nivel de la jerarquía de tópicos, a la carpeta donde se encuentra los ejecutables (CallPLSA.exe y PLSA4.exe). Como ejemplo se toma la jerarquía que se generó con un conjunto de 200 documentos, esta jerarquía es de tres niveles y en el segundo nivel se encuentran los archivos que corresponde a la matriz de términos por documento necesarios para hallar las probabilidades con PLSA (ver **Figura 114**).

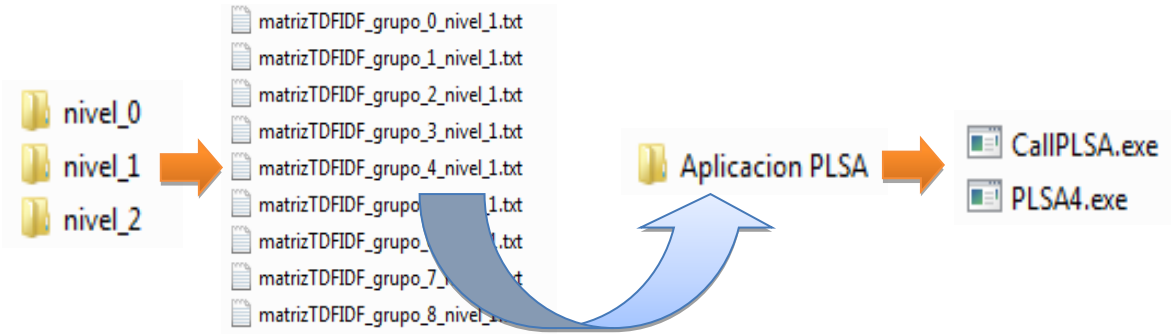


Figura 114. Copiar archivos correspondientes a la matriz de documento del penúltimo nivel en la carpeta de la aplicación de PLSA

- **Paso 2:** Ejecutar la aplicación CallPLSA.exe que usa PLSA4.exe, este último contiene el algoritmo PLSA con sus métodos para hallar las probabilidades (ver **Figura 115**, **Figura 116** y **Figura 117**).

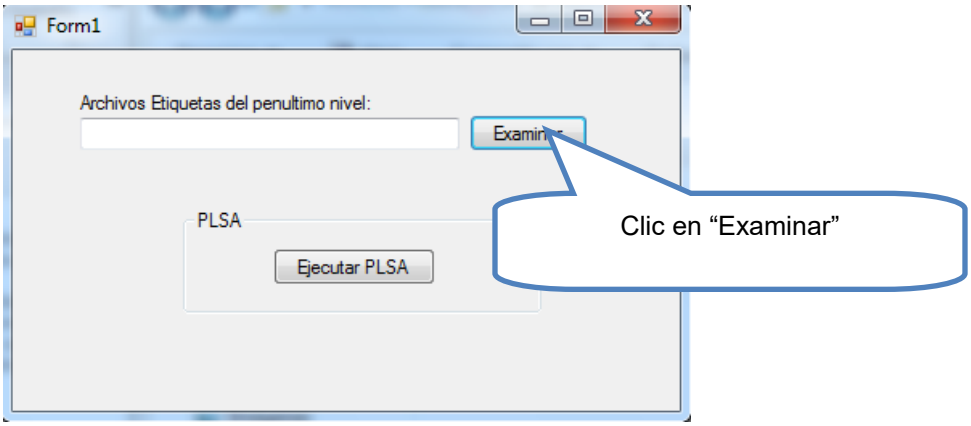


Figura 115. Aplicación que utiliza el algoritmo PLSA

[25] Por sus siglas en inglés Probabilistic Latent Semantic Analysis

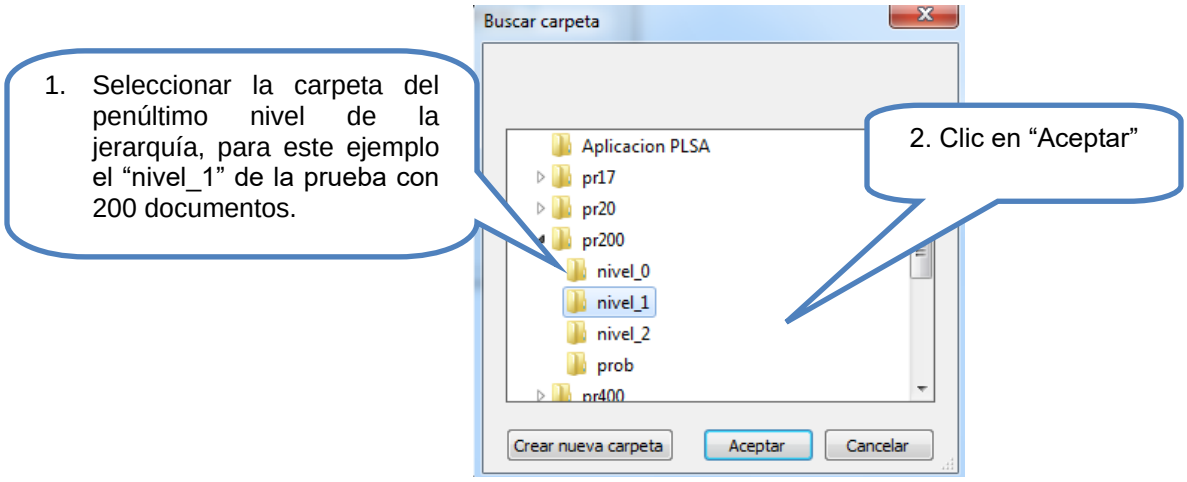


Figura 116. Seleccionar los archivos del penúltimo nivel de la jerarquía de los 200 documentos

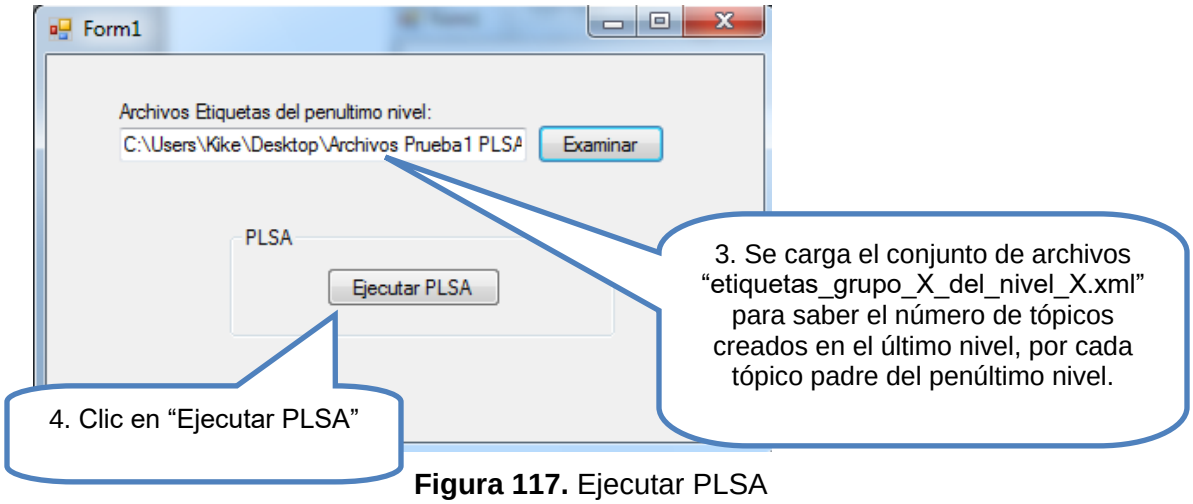


Figura 117. Ejecutar PLSA

- **Paso 3:** Al ejecutarse la aplicación, se generan un conjunto de archivos que contienen las probabilidades P(w|z) y P(z|d) por cada tópico del ultimo nivel de la jerarquía de tópicos, este conjunto de archivos se deben guardar en una carpeta con nombre "*prob*" que se usara en el algoritmo Cosme, como ejemplo se muestran los archivos (ver **Figura 118**) con las probabilidades para el conjunto de 200 documentos del ultimo nivel de la jerarquía de tópicos.

Figura 118. Archivos generados por la aplicación PLSA y guardados en la carpeta “prob”

ANEXO H – CREACIÓN Y CONFIGURACIÓN DE DUNDAS OLAP

"Dundas OLAP Services", es la herramienta que se usó para el despliegue de la Bodega de Documentos. A continuación se describe el proceso de desarrollo que se utilizó para crear la aplicación con Dundas. Los pasos a seguir son:

1. Crear un nuevo proyecto de Visual c#, con la plantilla "Aplicación web ASP.NET".

2. Crear en un nuevo TAB llamado "Dundas OLAP Services" y adicionar las librerías (DundasWebOlapDataProviderAdomd.dll, DundasOlapWebUIControls.dll, DundasWebOlapDataProviderADOMDNet.dll y DundasOlapManager.dll).

3. En Default.aspx se adicionan los controles (AdomdNetDataProvider y OlapClient) y posteriormente por un lado se configura la cadena de conexión en la cual se pone el nombre de la base de datos multidimensional y el nombre del equipo, por ejemplo (Data Source =TESIS; Initial Catalog = Proyecto de Analysis Services1) y por otro lado se configure la propiedad "DataProviderID" colocando el nombre del control "AdomdNetDataProvider".

En *References*, agregar las referencias (DundasOlapDataProvider y DundasWebOlapManager). En Web.configagregar la siguientesentencia:

```
<dependentAssembly>
        <assemblyIdentityname="Microsoft.AnalysisServices.AdomdClient"publicKeyToken="89845
dcd8080cc91"/>
        <bindingRedirectoldVersion="9.0.242.0"newVersion="10.0.0.0"/>
</dependentAssembly>
```

Si todo esta correcto se observa la interfaz de la **Figura 119**.

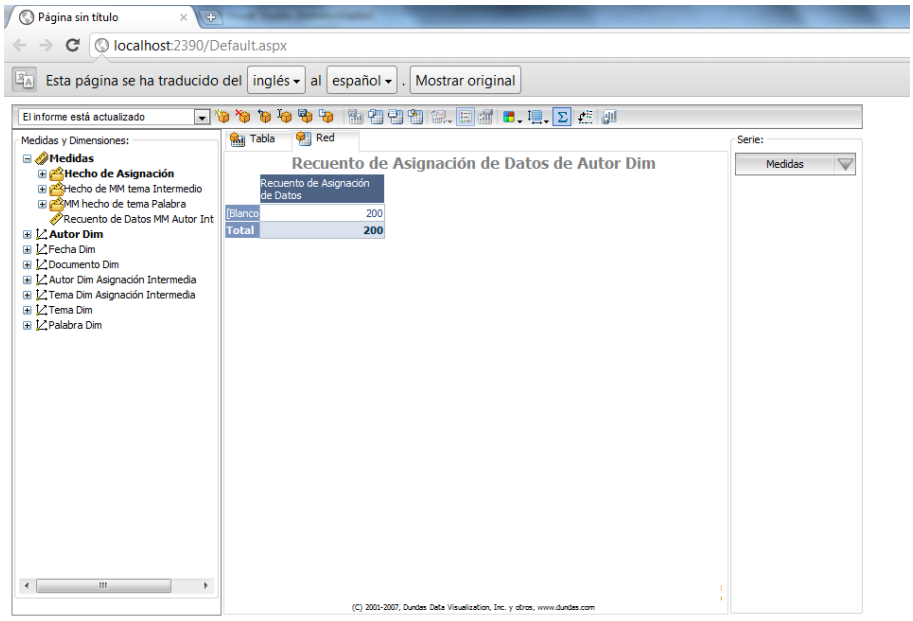


Figura 119. Herramientas Dundas OLAP

ANEXO I – IGBHSK MODIFICADO

El primer paso para la ejecución del algoritmo, es adjuntar la base de datos que usa el algoritmo para su funcionamiento, para esto se ejecuta Microsoft SQL Server Management Studio 2008 se hace el inicio de sesión ya sea con el usuario "*sa*" o con autenticación Windows. Ubicarse en Bases de Datos, clic con el botón derecho y en el menú contextual seleccionar "*adjuntar...*". Ver **Figura 120**.

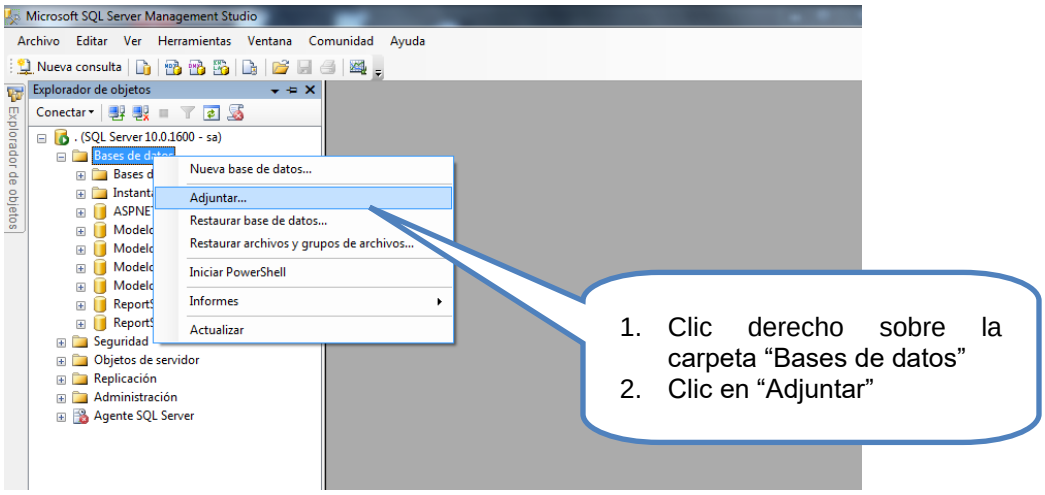


Figura 120. Microsoft Sql Server Management Studio – Adjuntar Base de Datos

En la ventana siguiente se escoge la base de datos que se desea adjuntar, para ello se da clic en el botón Agregar y en la nueva ventana que se despliega se ubica el archivo de base de datos del algoritmo IGBHSK modificado llamado "ASPNET…" como se muestra en la Figura 121, por último dar clic en aceptar.

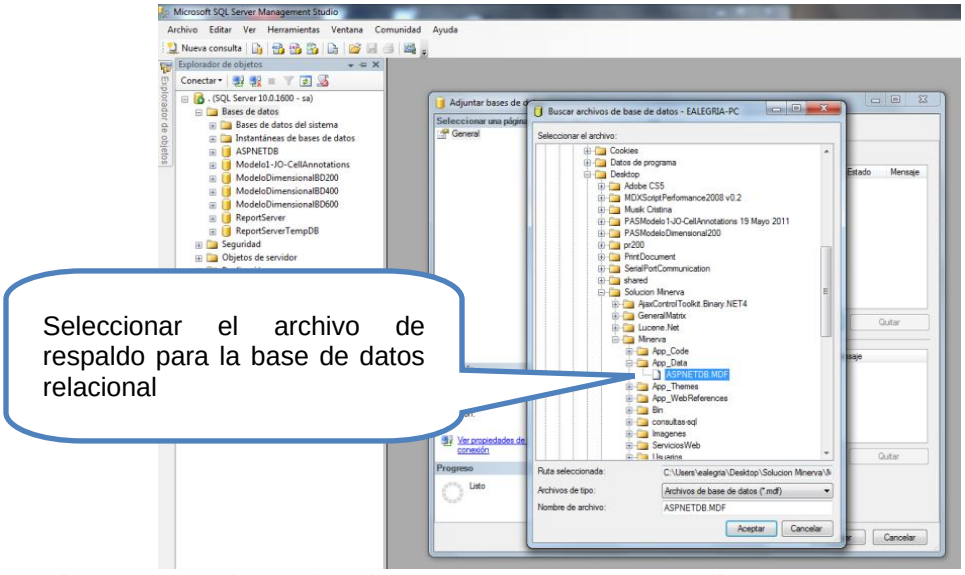


Figura 121. Microsoft Sql Server Management Studio – Agregar Base de Datos

A continuación se modifica el archivo web.config de la aplicación y la cadena de conexión con los datos del computador donde se esté montando la base de datos (ver **Figura 122**).

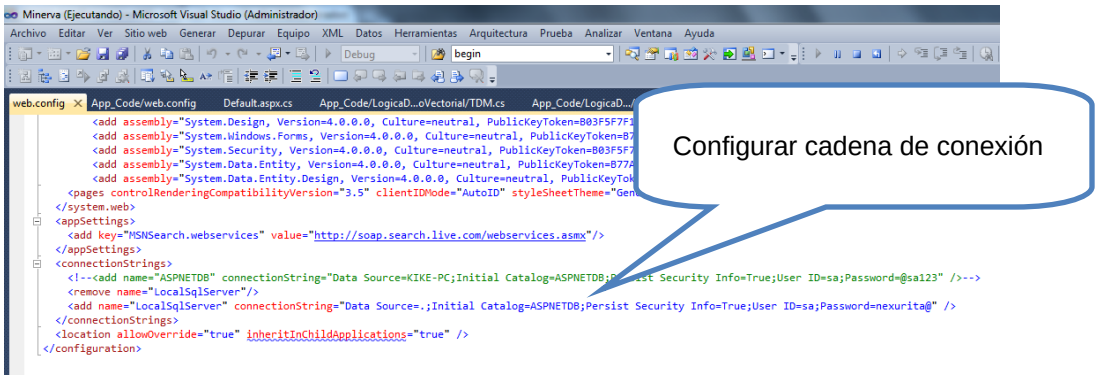


Figura 122. Modificación del archivo web-config

En esta cadena de conexión se coloca el ID y Password del motor de base de datos SQL Server 2008 a utilizar. Por último se debe asegurar que en la carpeta C:/Temp contenga el archivo con el conjunto de datos iniciales con el cual trabaja el algoritmo, este archivo debe llamarse: "docs_grupo_All_nivel_0.xml" y debe de contener la estructura mostrada en la **Figura 123**.

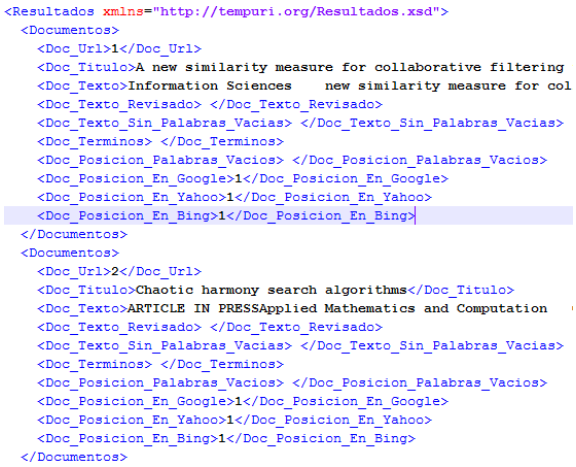


Figura 123. Estructura Conjunto de Datos inicial

Luego de esto se ejecuta la aplicación y se observa una interfaz web donde se encuentra un enlace "*Iniciar sesion*" (ver **Figura 124**) que pedirá un usuario y password del sistema, cuando se ejecuta por primera vez se debe crear un usuario, hacer clic en registrar (ver **Figura 125**), este enlace lleva a un formulario para ingresar un usuario a nuestro sistema.

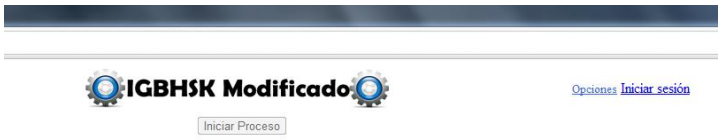


Figura 124. Pantalla Inicio IGBHSK Modificado

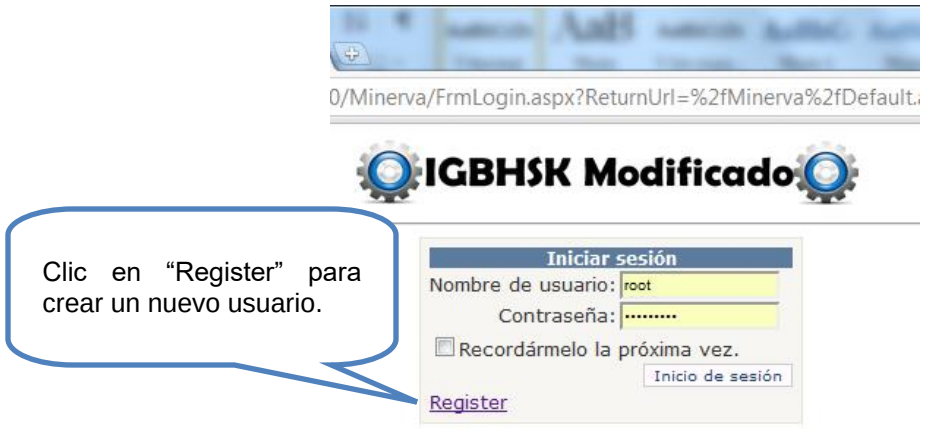


Figura 125. Inicio de Sesión Algoritmo IGBHSK Modificado

Luego se regresa al formulario de ingreso como se muestra en la Figura 124, se inicia sesión y se procede a iniciar el proceso, para ello se debe hacer clic en el botón "*Iniciar Proceso*". Ver **Figura 126**.

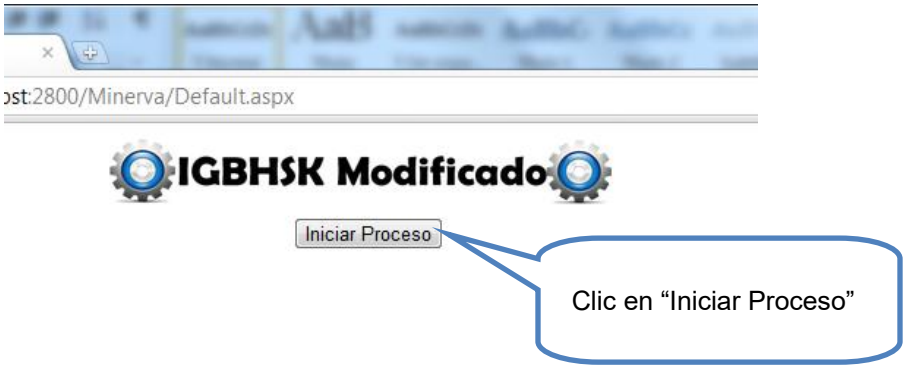


Figura 126. Inicio del proceso de IGBHSK Modificado

Cuando el algoritmo haya terminado, habrá generado una jerarquía de carpetas (C:/Temp) que indican la jerarquía de documentos junto con sus etiquetas y demás archivos necesarios para los siguientes procesos (ver **Figura 127**).

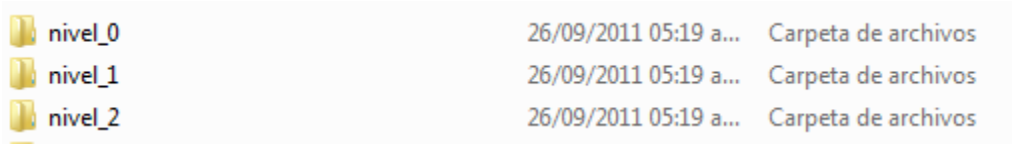


Figura 127. Estructura Jerarquía generada por IGBHSK Modificado

ANEXO J – ALGORITMO COSME

Configuración

Inicialmente se debe escoger las carpetas donde se guardaran los archivos generados "Ruta salida", la ruta donde se guardaran los documentos en formato de texto (TXT), y la ruta de los documentos de los en archivos pdf. Se debe de tener en cuenta que el paso uno de este algoritmo necesita del archivo export.txt el cual debe estar en la misma ruta de salida configurada anteriormente. Este archivo contiene los metadatos de todos el conjunto de documentos, ver Figura 128.

```
Reference Type:  Book Section
Record Number: 925
Author: A. Abraham, A.-E. Hassanien, P. Siarry, A. Engelbrecht and Z. Geem
Year: 2009
Title: Global Optimization Using Harmony Search: Theoretical Foundations and Applications
Book Title: Foundations of Computational Intelligence Volume 3
Publisher: Springer Berlin / Heidelberg
Volume: 203
Pages: 57-73
Series Title: Studies in Computational Intelligence
Short Title: Global Optimization Using Harmony Search: Theoretical Foundations and Applications
DOI: 10.1007/978-3-642-01085-9_3
Abstract: This chapter presents the theoretical foundations of the musicphenomenon- mimicking optimiz
URL: http://dx.doi.org/10.1007/978-3-642-01085-9_3
```

Figura 128. Metadatos del archivo export.txt

Una vez configurado estos parámetros desde el código se inicia la aplicación y se observa la siguiente pantalla:

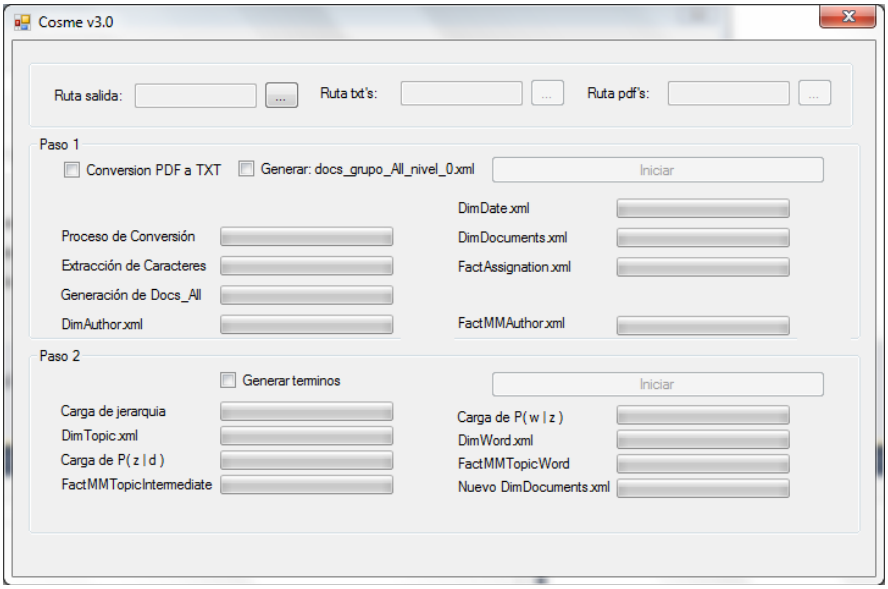


Figura 129. Interfaz Algoritmo Cosme

Ahora se selecciona "Conversión PDF a TXT" e iniciar el primer paso del algoritmo Cosme (ver Figura 129). Este paso se encarga de la conversión de los archivos pdf a archivos de texto (TXT), la extracción de caracteres extraños después del proceso de conversión, la generación del conjunto de datos inicial "docs_grupo_All_nivel_0.xml" solo si se selecciona esta opción, la generación de los archivos XML para las dimensiones Author, Date, Documents y la generación de dos archivos xml para las tablas de hecho FactMMAuthor y Fact Assignation.

El segundo paso de este algoritmo solo se debe realizar una vez haya terminado el primer paso y se tenga los resultados del algoritmo PLSA, el cual debe proporcionar una carpeta con la información de las probabilidades para cada tópico, esta carpeta debe de llamarse "*prob*" y debe estar ubicada en la misma carpeta de salida configurada inicialmente. El paso dos de Cosme carga la jerarquía generada por IGBHSK modificado y las probabilidades de P(z|d) y P(w|z) para generar los archivos para la dimensión Topic y la dimensión Word, los archivos para las tablas de hecho FactMMTopicIntermediate, FactMMTopicWord y la modificación del archivo para la dimensión Documents. Debe de tenerse en cuenta seleccionar la opción "Generar Términos".

ANEXO K – FORMATO ENCUESTA

ENCUESTA, MODELO DE ANALISIS MULTI-DIMENSIONAL PARA UNA BODEGA DE DOCUMENTOS QUE INCLUYE MEDIDAS TEXTUALES

Objetivo:
EL objetivo de esta encuesta es evaluar la satisfacción del usuario al navegar por las jerarquías de las dimensiones y por las medidas textuales de la bodega de documentos propuesta, con respecto a la facilidad de uso y al tiempo de ejecución de la consulta.

Guía
Las consultas se presentan a continuación:

- Una dimensión:
  - Consulta 1: La jerarquía de la Dimensión *Document* junto con la medida de texto Topics_Probab.
  - Consulta 2: La jerarquía de la Dimensión Date junto con la medida de texto Topics_Probab.
  - Consulta 3: La jerarquía de la dimensión Topic junto con la medida de texto Documents.

- Dos Dimensiones:
  - Consulta 4: La combinación de las jerarquías de Document y Date junto con la medida de texto Topics_probab.

Dirigido a:
La población objetivo es un grupo de tesistas del programa de ingeniería de sistemas de la Universidad del Cauca.

Encuestadores:

- ✓ Manuel Enrique Maca Orozco
- ✓ Erwin Martin Alegría Velásquez

Fecha de encuesta: _____

Género: ☐ Masculino ☐ Femenino **Nro. Encuesta:** _____

Opciones de respuesta y Observación:

En las respuestas Totalmente en desacuerdo, en desacuerdo y ni de acuerdo ni en desacuerdo especificar el porqué de esta respuesta sobre la línea punteada de cada pregunta.

[1]. Totalmente en desacuerdo
[2]. En desacuerdo
[3]. Ni acuerdo ni en desacuerdo
[4]. De acuerdo

[5]. Totalmente de acuerdo

- ✓ **Factor: Facilidad de Uso**

1. El sistema presenta una fácil navegación?

Consulta 1: [1] [2] [3] [4] [5]
Porque?_____

Consulta 2: [1] [2] [3] [4] [5]
Porque?_____

Consulta 3: [1] [2] [3] [4] [5]
Porque?_____

Consulta 4: [1] [2] [3] [4] [5]
Porque?_____

2. Es entendible la presentación de la medida textual para el análisis?

Consulta 1: [1] [2] [3] [4] [5]
Porque?_____

Consulta 2: [1] [2] [3] [4] [5]
Porque?_____

Consulta 3: [1] [2] [3] [4] [5]
Porque?_____

Consulta 4: [1] [2] [3] [4] [5]
Porque?_____

3. El formato de presentación de la medida textual es adecuado?

Consulta 1: [1] [2] [3] [4] [5]
Porque?_____

Consulta 2: [1] [2] [3] [4] [5]
Porque?_____

Consulta 3: [1] [2] [3] [4] [5]
Porque?_____

Consulta 4: [1] [2] [3] [4] [5]
Porque?_____

4. La información presentada por la medida textual satisface sus expectativas?

Consulta 1: [1] [2] [3] [4] [5]
Porque?_____

Consulta 2: [1] [2] [3] [4] [5]

Porque?_____

Consulta 3: [1] [2] [3] [4] [5]
Porque?_____

Consulta 4: [1] [2] [3] [4] [5]
Porque?_____

5. Es entendible la relación entre la medida textual y la(s) jerarquía(s) de la(s) dimensión(es)?

Consulta 1: [1] [2] [3] [4] [5]
Porque?_____

Consulta 2: [1] [2] [3] [4] [5]
Porque?_____

Consulta 3: [1] [2] [3] [4] [5]
Porque?_____

Consulta 4: [1] [2] [3] [4] [5]
Porque?_____

✓ **Factor: Tiempo de Consulta**

6. El tiempo de respuesta fue lo suficientemente rápido?

Consulta 1: [1] [2] [3] [4] [5]
Porque?_____
Consulta 2: [1] [2] [3] [4] [5]
Porque?_____

Consulta 3: [1] [2] [3] [4] [5]
Porque?_____

Consulta 4: [1] [2] [3] [4] [5]
Porque?_____

7. El tiempo de respuesta fue el esperado?

Consulta 1: [1] [2] [3] [4] [5]
Porque?_____

Consulta 2: [1] [2] [3] [4] [5]
Porque?_____

Consulta 3: [1] [2] [3] [4] [5]
Porque?_____

Consulta 4: [1] [2] [3] [4] [5]
Porque?_____

8. El tiempo de respuesta de las consultas se ajusta a sus expectativas?

Consulta 1: [1] [2] [3] [4] [5]
Porque?_____

Consulta 2: [1] [2] [3] [4] [5]
Porque?_____

Consulta 3: [1] [2] [3] [4] [5]
Porque?_____

Consulta 4: [1] [2] [3] [4] [5]
Porque?_____

- ✓ **Concepto general:**

9. En su percepción, la navegación en el modelo de documentos a través de las medidas textuales es útil en el caso de una revisión bibliográfica?

Consulta 1: [1] [2] [3] [4] [5]
Porque?_____

Consulta 2: [1] [2] [3] [4] [5]
Porque?_____
Consulta 3: [1] [2] [3] [4] [5]
Porque?_____

Consulta 4: [1] [2] [3] [4] [5]
Porque?_____

ANEXO L – PROCESO ETL TABLAS DE HECHO Y DE DIMENSIÓN

A continuación se describe el proceso que se realizó para el proceso ETL de la bodega de documentos

TABLAS DE DIMENSION

Como se ha mencionado, los primeros datos que fueron cargados son los correspondientes a las tablas de dimensión. Con base en el modelo lógico (ver Figura 22), se inició la carga con las dimensiones más simples a las más complejas de acuerdo a la integración de datos a partir de paquetes en la herramienta SQL Server Integration Services 2008.

- **DimIntermAssignationTopic:** como solo tiene un atributo, el cual es su llave primaria, los datos o valores correspondientes a este atributo son igual al número de documentos que se han procesado, para este proyecto se utilizaron 200, 400 y 600 documentos científicos.

Atributo	Transformación Tipo
IntermediateATKey (Primary Key)	Entero de cuatro bits con signo [DT_I4]

- **DimIntermAssignationAuthor:** al igual que la dimensión anterior solo tiene un atributo.

Atributo	Transformación Tipo
IntermediateAAKey (Primary Key)	Entero de cuatro bits con signo [DT_I4]

Los archivos que se utilizaron para cargar las siguientes dimensiones fueron generadas por el algoritmo Cosme en el paso 2:

- **DimWord:** corresponde al archivo *DimWord.xml*, el cual contiene las palabras correspondientes a cada tópico.

Atributo	Transformación Tipo
WordKey (Primary Key)	La llave se genera automáticamente en la base de datos relacional
Word	Cadena [DT_STR], termino o palabra

- **DimTopic:** corresponde al archivo *DimTopic.xml*, el cual contiene los tópicos con su estructura jerárquica (padre-hijo).

Atributo	Transformación Tipo
TopicKey (Primary Key)	La llave se genera automáticamente en la base de datos relacional
Topic	cadena [DT_STR], tópico
ParentTopicKey	Entero de cuatro bits con signo [DT_I4], correspondiente a la llave primaria de su tópico padre

Los archivos que se utilizaron para cargar las siguientes dimensiones fueron generadas por el algoritmo Cosme en el paso 1:

- **DimDocument:** corresponde al archivo *DimDocuments.xml*, el cual contiene la meta datos de los documentos científicos procesados.

Atributo	Transformación Tipo
DocumentKey (Primary Key)	La llave se genera automáticamente en la base de datos relacional
NameDocument	Cadena [DT_STR], nombre del documento
TypeDocument	cadena [DT_STR], tipo de documento
Topics	Cadena [DT_STR], tópico(s) que menciona el documento

- **DimAuthor:** corresponde al archivo *DimAuthor.xml*, el cual contiene la información de los autores de los documentos científicos.

Atributo	Transformación Tipo
AuthorKey (Primary Key)	La llave se genera automáticamente en la base de datos relacional
Name	Cadena [DT_STR], nombre del autor
LastName	Cadena [DT_STR], apellido del autor
Email	Cadena [DT_STR], correo electrónico del autor

- **DimDate:** corresponde al archivo *DimDate.xml*, el cual contiene las fechas en las cuales fueron publicados los documentos científicos.

Atributo	Transformación Tipo
DateKey (Primary Key)	La llave se genera automáticamente en la base de datos relacional
FullDate	Cadena [DT_STR], fecha completa de publicación
Year	Entero de cuatro bits con signo [DT_I4], correspondiente al año de publicación
Month	Entero de cuatro bits con signo [DT_I4], correspondiente al mes de publicación
Day	Entero de cuatro bits con signo [DT_I4], correspondiente al día de la publicación

TABLAS DE HECHO

Después de tener los datos cargados en las dimensiones, se procede a cargar los datos en las dimensiones de hecho.

- **FactMMAuthorIntermediate:** corresponde al archivo FactMMAuthor.xml generado por el algoritmo Cosme en el paso 1, el cual contiene las relaciones entre las dimensiones DimAuthor y DimIntermAssignationAuthor.

Atributo	Transformación Tipo
IntermediateAAKey (Foreign Key)	Entero de cuatro bits con signo [DT_I4], correspondiente a la llave foránea de la dimensión DimIntermAssignationAuthor.
AuthorKey (Foreign Key)	Entero de cuatro bits con signo [DT_I4], correspondiente a la llave foránea del autor

- **FactMMTopicWord:** corresponde al archivo *FactMMTopicWord.xml* generado por el algoritmo Cosme en el paso 2, el cual contiene las relaciones entre las dimensiones *DimWord* y *DimTopic,* y la probabilidad de la palabra en el tópico (*pWordTopic*).

Atributo	Transformación Tipo
TopicKey (Foreign Key)	Entero de cuatro bits con signo [DT_I4], correspondiente a la llave foránea del tópico
WordKey (Foreign Key)	Entero de cuatro bits con signo [DT_I4], correspondiente a la llave foránea de la palabra o término
pWordTopic	Numérico [DT_NUMERIC], correspondiente a la probabilidad de la palabra en el tópico

- **FactMMTopicIntermediate:** corresponde al archivo *FactMMTopicIntermediate.xml* generado por el algoritmo Cosme en el paso 2, el cual contiene las relaciones entre las dimensiones *DimTopic* y *DimIntermAssignationTopic*, y la probabilidad del topico en el documento (*pTopicDoc*).

Atributo	Transformación Tipo
IntermediateATKey (Foreign Key)	Entero de cuatro bits con signo [DT_I4], correspondiente a la llave foránea de la *DimIntermAssignationTopic*
TopicKey (Foreign Key)	Entero de cuatro bits con signo [DT_I4], correspondiente a la llave foránea del tópico
pWordTopic	Numérico [DT_NUMERIC], correspondiente a la probabilidad del tópico en el documento

- **FactAssignation:** corresponde al archivo *FactAssignation.xml* generado por el algoritmo Cosme en el paso 1, el cual contiene las relaciones entre las dimensiones *DimDocument, DimAuthor, DimDate*y *DimTopic*, centrando en ella las medidas textuales.

Atributo	Transformación Tipo
DocumentKey (Foreign Key)	Entero de cuatro bits con signo [DT_I4], correspondiente a la llave foránea del documento
DateKey (Foreign Key)	Entero de cuatro bits con signo [DT_I4], correspondiente a la llave foránea de la fecha
IntermediateAAKey (Foreign Key)	Entero de cuatro bits con signo [DT_I4], correspondiente a la llave foránea de la *DimIntermAssignationAuthor*
IntermediateATKey (Foreign Key)	Entero de cuatro bits con signo [DT_I4], correspondiente a la llave foránea de la *DimIntermAssignationTopic*

ANEXO M – ARTICULO