



# *Schema Workbench*

*Tecnologías de la Información para  
las Organizaciones*

Josué Toledo – Carlota Lázaro

## Introducción

El presente informe tiene como objetivo realizar una presentación acerca de la herramienta *Schema Workbench* que se encuentra disponible en la suite *Pentaho* para la asignatura *Tecnologías de la Información para las Organizaciones*. Dicha herramienta es una interfaz de diseño que te permite crear y probar esquemas de cubos OLAP Mondrian visualmente.

## Glosario de términos

**Data Warehousing(DW):** En el contexto de la informática, un almacén de datos (del inglés *data warehouse*) es una colección de datos orientada a un determinado ámbito (empresa, organización, etc.), integrado, no volátil y variable en el tiempo, que ayuda a la toma de decisiones en la entidad en la que se utiliza.

**Business Intelligence:** Conjunto de estrategias y aspectos relevantes enfocados a la administración y creación de conocimiento sobre el medio, a través del análisis de los datos existentes en una organización o empresa.

**Pentaho:** Pentaho BI Suite es un conjunto de programas libres para generar inteligencia empresarial. Incluye herramientas integradas para generar informes, minería de datos, ETL, etc.

## Instalación

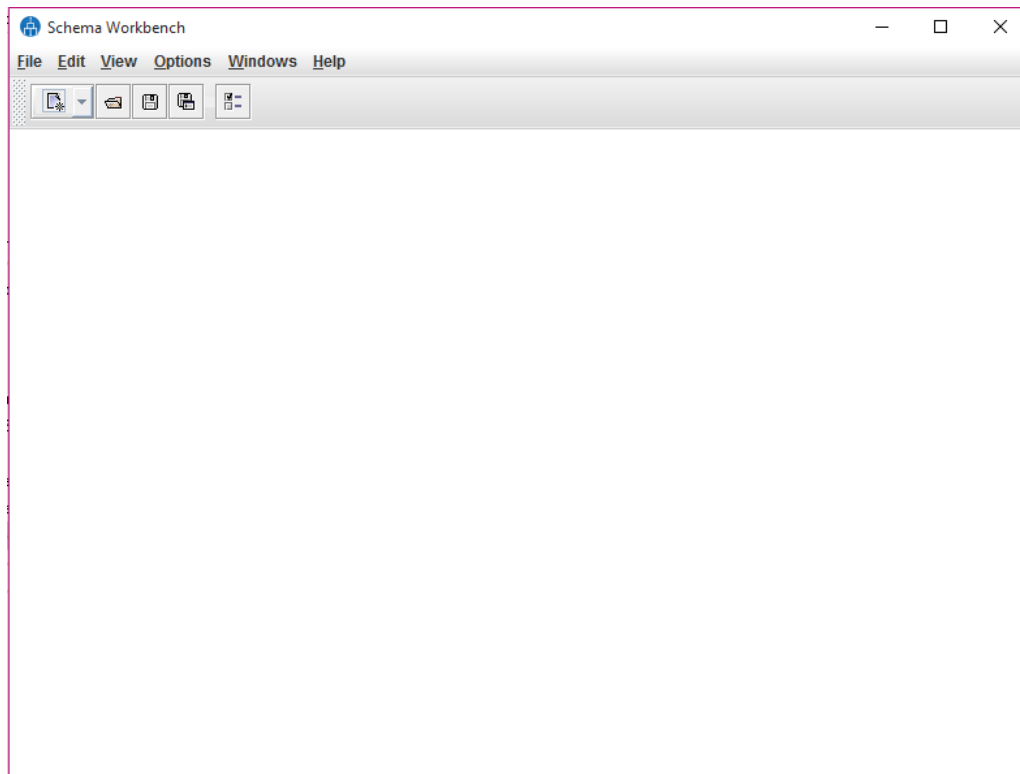
Para la instalación de la herramienta *Schema Workbench*, hemos accedido a la web oficial de *Pentaho* (<http://community.pentaho.com/>) y, hemos descargado el software haciendo click en el enlace.

Una vez descargado, procedemos a la instalación de la herramienta. Para ello, hemos descomprimido el programa y accedido al directorio *schema-workbench*, y finalmente ejecutado el fichero *workbench* que se trata de un archivo de procesamiento por lotes, distinguiéndose dos ejecutables en función del sistema operativo en el que se desea implantar:

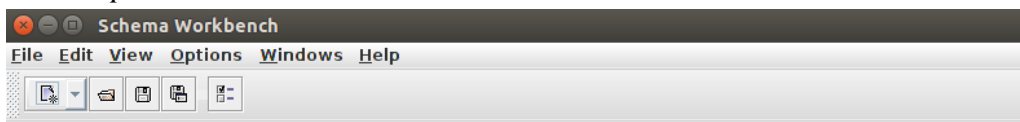
1. Fichero *workbench.bat* : Cuando se desea utilizar *Schema Workbench* en Windows.
2. Fichero *workbench.sh* : Cuando se desea utilizar *Schema Workbench* en Linux.

Una vez realizados los pasos anteriores, se presentará la siguiente interfaz donde comenzaremos a trabajar con la aplicación:

*Interfaz de la aplicación en Windows*



*Interfaz de la aplicación en Linux:*



---

Como podemos comprobar son bastantes similares los diseños de la interfaz en los distintos sistemas operativos, presentando algunas pequeñas diferencias únicamente en lo que respecta a colores o tipo de letra empleada.

Cabe destacar que, para poder hacer uso del software debemos tener instalado en nuestro sistema Java.

### *Características principales*

Pentaho Schema Workbench es una herramienta de desarrollo que permite crear, modificar y publicar un esquema de Mondrian para que sean utilizados en análisis por los usuarios de la plataforma Pentaho. Se encuentra muy orientado a aquellos desarrolladores que conocen la estructura de un esquema de este tipo, que presenta las siguientes características entre otras:

1. Esquema que define una base de datos multidimensional.
2. Contiene un modelo lógico que consiste en las estructuras usadas para escribir consultas en lenguaje MDX.
3. El modelo lógico se mapea en un modelo físico, el cual representa las tablas de datos presentadas previamente en el modelo lógico.
4. Son archivos de tipo xml(pueden editarse manualmente o a través de Schema Workbench de Pentaho).

La herramienta Schema Workbench permite crear todos los objetos que soporta Mondrian: esquema, cubo, dimensiones, métricas,... Otras características relevantes de este software son las siguientes:

1. Es un programa java multiplataforma.
2. Se requieren conocimientos en DW(data warehousing) y ambientes OLAP.
3. Puede explotar todas las características del motor OLAP, al menos lo que concierne a diseño o modelado.
4. Actualmente Pentaho promociona(incluso en sus cursos oficiales) a Schema Workbench como la herramienta adecuada para la creación de cubos.
5. Permite realizar consultas MDX sobre el esquema creado(requiere conocer la sintaxis del lenguaje).
6. Permite consultar la base de datos que sirve de origen para el esquema de Mondrian.
7. Permite publicar directamente el esquema en el servidor de Pentaho.
8. Genera ficheros XML editables, los cuales pueden modificarse para añadir elementos como “key expression” u otro tipo de condiciones o restricciones con el objetivo de que se incorporen en el schema.

### *Complejidad y problemas frecuentes*

En este caso, dado que el Sistema Operativo contenía todos los componentes necesarios para la instalación de la aplicación y su correcto funcionamiento, no se ha llegado a dar una gran complejidad en lo que respecta a la instalación y ejecución del software.

En el ámbito del funcionamiento general de la aplicación y el desarrollo de proyectos, algunos de los problemas más frecuentes que se pueden generar con la herramienta son los siguientes:

1. Conexión errónea con la base de datos a partir de la cual vamos a generar y construir el esquema Mondrian. Antes de iniciar cualquier proceso, es necesario vincular dicho esquema con los datos correspondientes. Una vez que se establece la conexión, se realiza un testeo de la misma para comprobar que realmente está operativa y funcionando correctamente.
2. Vinculación errónea de las dimensiones con el cubo, bien sea porque intentamos asociar dimensiones de un cubo con las de otro o bien porque no establecemos correctamente las claves ajenas con lo cual no puede entablarse comunicación entre las mismas.
3. Las dimensiones deben contener las tablas necesarias para que puedan hacerse las mediciones o filtraciones correctamente.
4. Haciendo referencia a las habilidades de programación de los distintos usuarios, cabe destacar que la introducción de sentencias erróneas en el fichero XML puede ocasionar problemas o inconsistencias en el esquema, pudiendo no ser interpretadas correctamente por Schema Workbench.

### *Definición del módulo*

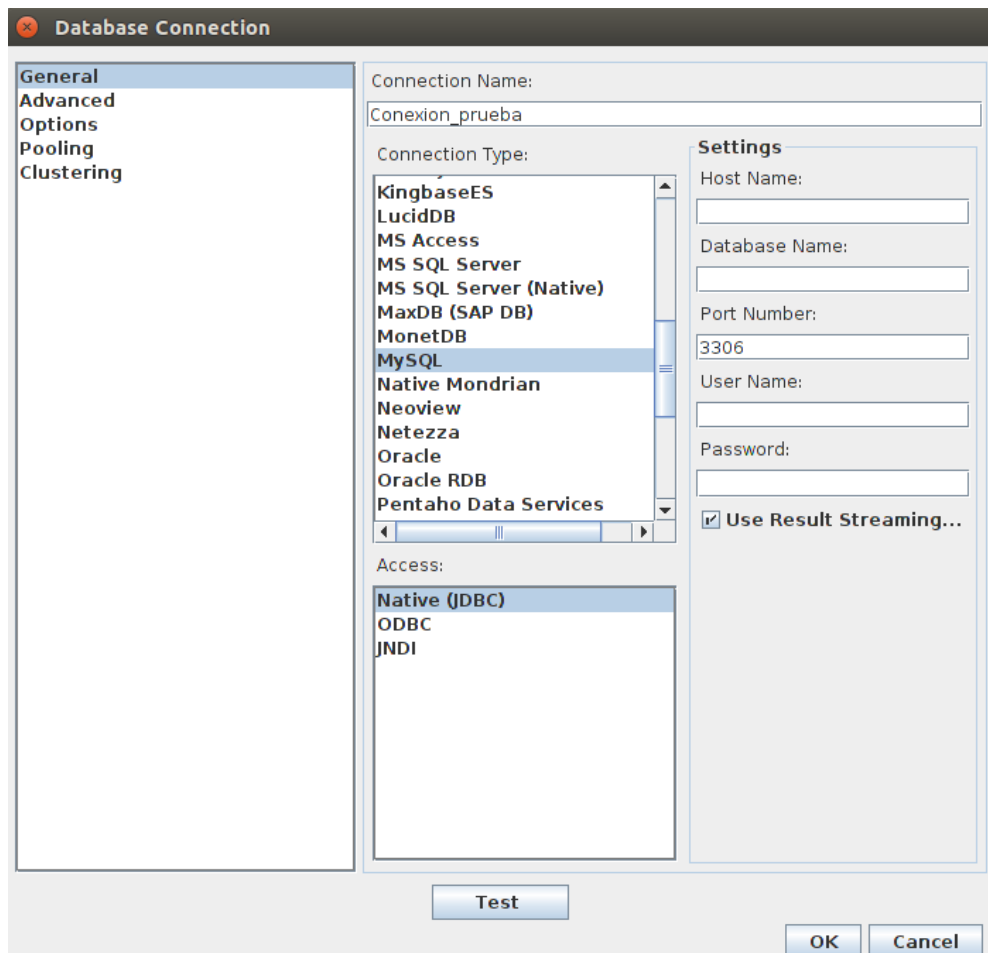
Para la utilización de *Schema Workbench*, al igual que todas las aplicaciones desarrolladas por *Pentaho*, deben definirse en primer lugar las conexiones a la base de datos como paso previo a la configuración de los esquemas y además, se debe configurar la API que se usará para posibilitar la realización de operaciones sobre los datos.

Antes de hacer referencia al funcionamiento de la herramienta, es preciso conocer algunos conceptos y elementos importantes que utilizaremos para desarrollar esquemas:

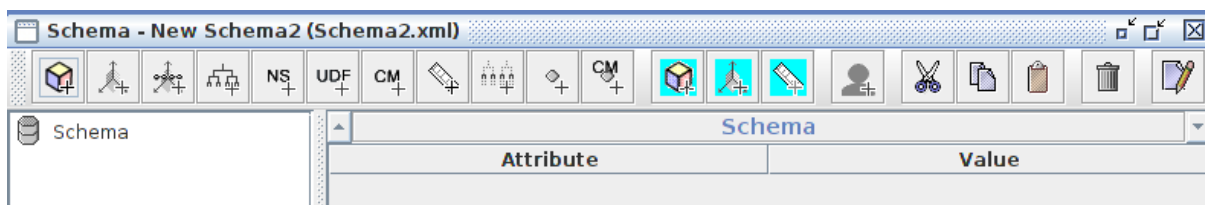
1. Cubo: Puede definirse como una estructura multidimensional de información, que puede contener dimensiones, métricas, miembros calculados,... Contiene los datos de una base de datos o de sistemas transaccionales.
2. Dimensión: ejes de análisis o variables por las que los usuarios podrán filtrar los datos. Dentro de una dimensión puede definirse una jerarquía de elementos condicionando así el modo de representación de la información. Cada dimensión debe estar asociada a una tabla de la que se extrae información, mientras que las jerarquías deben vincularse a columnas de dicha tabla.
3. Cubos/Dimensiones virtuales: Son cubos o dimensiones que se crean a partir de la información contenida en otros cubos o dimensiones, respectivamente. Schema Workbench permite la creación de este tipo de elemento, mientras que en otras herramientas con funcionalidades similares como Cube Designer no es posible.
4. Métricas:
5. Miembros calculados:

En términos generales, los pasos a seguir cuando utilizamos Schema Workbench en un proyecto son los siguientes:

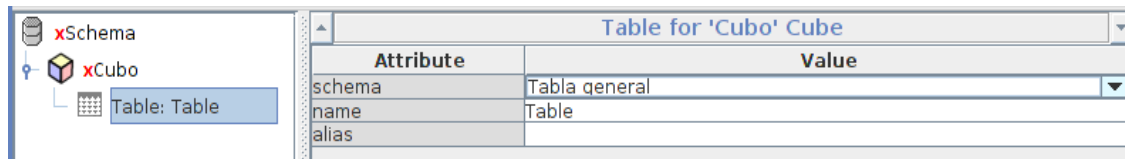
1. Conexión con base de datos (Mysql, Oracle, ...).



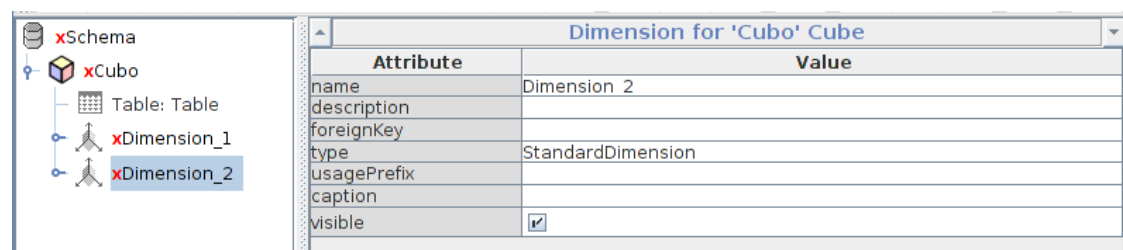
2. Se verifica que la conexión de la aplicación con la base de datos funciona correctamente a través del botón “Test” situado en el margen inferior de la captura de pantalla anterior.
3. Una vez que se ha verificado la conexión, configuramos la API y los drives necesarios para poder realizar operaciones sobre la base de datos. En este caso hemos configurado JDBC.
4. El paso siguiente consiste en la creación de un nuevo esquema. Para ellos hacemos click en el botón “New schema”.



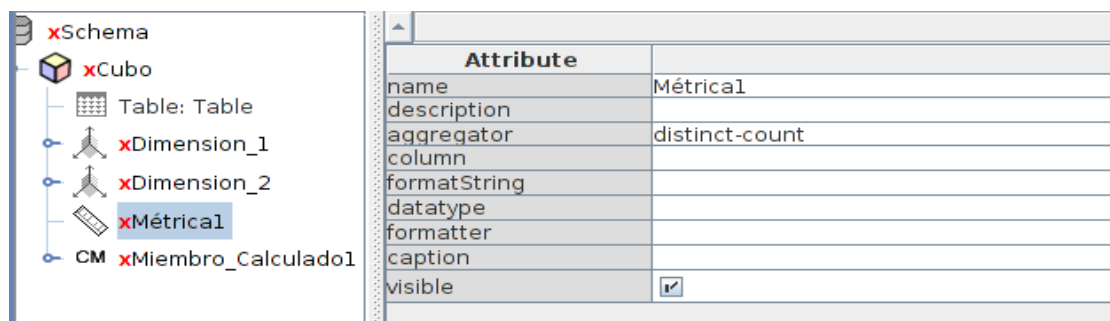
5. Posteriormente, construimos un nuevo cubo dentro del esquema y le asociamos una tabla de hechos (tabla principal de mi esquema), es decir, tabla que va a contener las variables sobre las cuales queremos información.



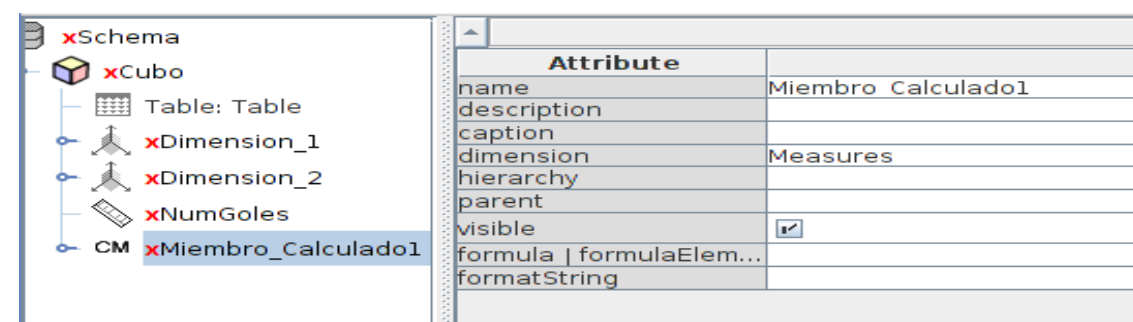
6. Una vez creado el cubo, podremos definir los siguientes elementos:
- Dimensiones privadas (dimensiones que no son compartidas por otros cubos).
  - Dimensiones públicas (dimensiones que son compartidas con otros cubos)



c. Métricas.



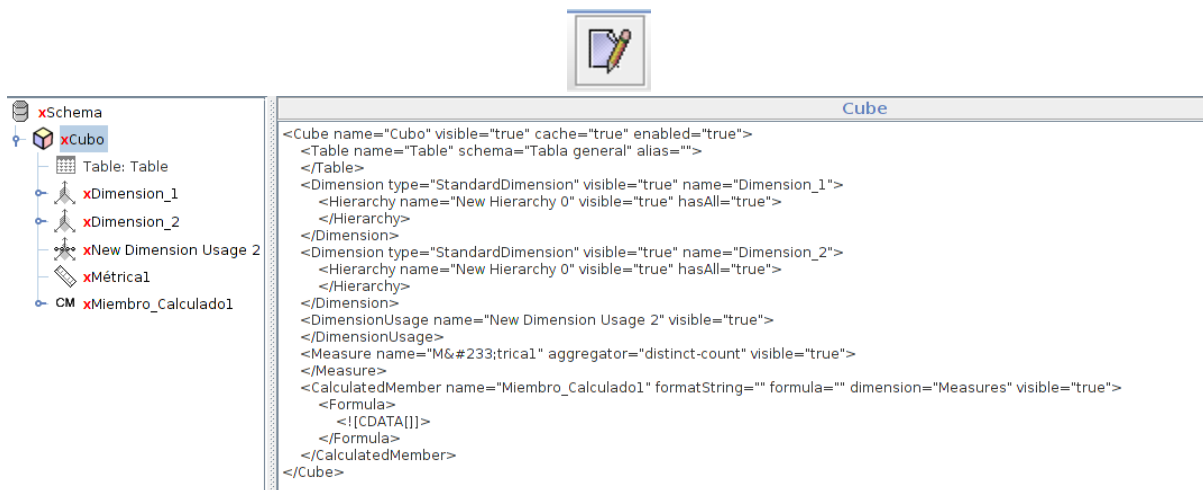
d. Miembros calculados.



Una vez creados estos elementos, debemos tener algunas consideraciones:

- Solo los cubos que se definen dentro de un mismo esquema pueden compartir las dimensiones.

2. Para que las dimensiones puedan relacionarse con el contenido del cubo es necesario crear las dimensiones haciendo click con el botón derecho del ratón sobre el mismo en la opción “Add Dimension Usage”, indicando que se establecerá una clave ajena en la nueva dimensión a través de la cual se producirán las comunicaciones.
3. Una vez que obtenemos el esquema final, guardamos y se crea un nuevo archivo en formato xml en el directorio de trabajo. Éste nuevo xml será editable por parte de los usuarios.
4. Podemos observar cómo se va codificando el archivo xml a medida de que añadimos elementos a nuestro esquema haciendo click en el siguiente botón del menú de edición de las propiedades de cada elemento:



### Principales elementos

Los principales elementos que presenta la plataforma son los siguientes:

- *Schema*: En el área de las Bases de Datos, entendemos por schema o esquema la estructura de una base de datos, en un lenguaje formal soportado por un *Sistema de Gestión de Base de Datos (DBMS)*.
- *XML*: Representan las siglas de *Extensible Markup Language*. Es un lenguaje de marcas desarrollado por el World Wide Web Consortium (W3C) utilizado para almacenar datos en forma legible.
- *OLAP*: Es una solución utilizada en el campo de la llamada Inteligencia de negocios (o *Business Intelligence*) cuyo objetivo es agilizar la consulta de grandes cantidades de datos. Para ello utiliza estructuras multidimensionales (o Cubos OLAP) que contienen datos resumidos de grandes Bases de datos o Sistemas Transaccionales (OLTP).



- *MDX*: es un lenguaje de consulta para bases de datos multidimensionales sobre cubos OLAP, se utiliza en Business Intelligence para generar reportes para la toma de decisiones basados en datos históricos, con la posibilidad de cambiar la estructura o rotación del cubo.
- *JDBC*: es una API que permite la ejecución de operaciones sobre bases de datos desde el lenguaje de programación Java, independientemente del sistema operativo donde se ejecute o de la base de datos a la cual se accede, utilizando el dialecto SQL del modelo de base de datos que se utilice.

### *Tecnologías que Integra*

Las tecnologías que integra la herramienta a analizar son las siguientes:

- Java (JRE): (Java Runtime Environment, o Entorno en Tiempo de Ejecución de Java) es el software necesario para ejecutar cualquier aplicación desarrollada para la plataforma Java.
- JDBC: colección de interfaces Java y métodos de gestión de manejadores de conexión hacia cada modelo específico de base de datos.
- Jpivot: herramienta de análisis de usuario, con el que realizamos la navegación dimensional sobre los cubos desde la plataforma BI y visualizamos los resultados de las consultas.

### *Análisis de su Interfaz*

A simple vista, la interfaz del software se caracteriza por sencillez e intuitividad. Los elementos se muestran con iconos que hacen posible una búsqueda rápida de aquello que es necesario usar en un momento determinado, por lo que se convierte en una herramienta de fácil aprendizaje.

Los elementos se encuentran colocados de forma coherente, comportándose de igual manera, además, los títulos de los campos reflejan el comportamiento que tendrá el evento seleccionado.

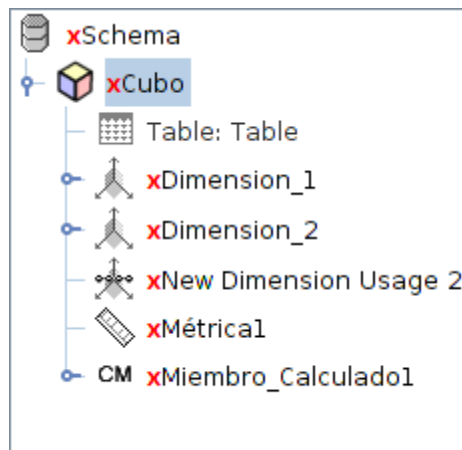
Por otra parte, el software presenta unos tipos de fuente legibles y hace uso de colores apropiados para la transmisión de información.

Por lo tanto, como conclusión podemos decir que estamos ante una herramienta que sigue los criterios de usabilidad.

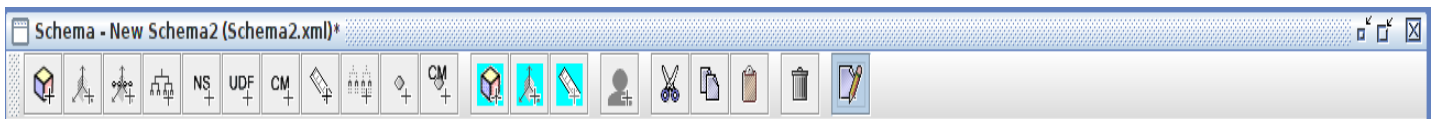
Sin embargo, la calidad de las imágenes instanciadas en el software no son de una gran calidad, por lo tanto, el software no genera una buena apariencia tal y como estamos hoy en día acostumbrados a ver en aplicaciones de escritorio, aplicaciones web, aplicaciones móviles, etc..

Estructuralmente presenta dos áreas principales y diferenciadas:

1. La zona en la que se muestra la estructura jerárquica del esquema OLAP.



2. La zona de edición de las propiedades de cada elemento. Presenta un menú superior para crear cubos, dimensiones, dimensiones conformadas, métricas, miembros calculados, subconjuntos (named set) y roles, así como operaciones estándar como cortar, copiar y pegar.



### *Software similar*

Existen aplicaciones similares a Schema Workbench destinadas a la creación de esquemas, es decir, de estructuras lógicas de una base de datos existentes expresadas en un lenguaje formal. Un ejemplo de aplicación similar puede ser Cube Designer, la cual presenta las siguientes características:

1. Sencillo asistente de 5 pasos, muy intuitivo. No se requieren grandes conocimientos en DW(Data warehousing) o ambientes OLAP para su uso.
2. Se caracteriza por una interfaz amigable para los usuarios, con un uso más importante de los colores en comparación con Schema Workbench.
3. Resulta especialmente útil cuando se pretende construir y configurar cubos de información sencillos.
4. No permite explotar todas las capacidades del motor OLAP. No pueden usarse cubos virtuales, roles,...
5. Su uso está decayendo en los últimos tiempos, notándose sobre todo en una menor existencia de documentación.

La gran diferencia con respecto a Schema Workbench es la utilización y aprovechamiento de los recursos del motor OLAP, superando éste a Cube Designer en gran medida y convirtiéndose, por lo tanto, en una herramienta más útil y productiva.

A continuación podemos observar la interfaz de este software similar. Concretamente se pueden visualizar las relaciones entre las distintas tablas de la base de datos:

