

# Implementación de Cubo OLAP con SQL Server

## 1. Descripción del Proyecto

**Objetivo:** Diseño, implementación y explotación de un Cubo de Datos (OLAP) utilizando tecnologías de Microsoft para el análisis multidimensional de información de ventas.

**Alcance:** El proyecto permite analizar métricas clave (Dinero y Cantidad) cruzando información por dimensiones geográficas (País, Región) y de catálogo (Producto, Categoría). Esto facilita la toma de decisiones estratégicas mediante consultas MDX avanzadas.

## 2. Arquitectura General

La solución sigue una arquitectura "On-Premise" utilizando el stack tecnológico de Microsoft:

- **Base de Datos Relacional:** SQL Server 2019 Developer Edition (Motor de Base de Datos).
- **Modelo de Datos:** Data Warehouse (Esquema de Estrella).
- **Motor OLAP:** SQL Server Analysis Services (SSAS) - Configurado en Modo Multidimensional.
- **Diseño del Cubo:** Visual Studio 2019 Community con extensión "Microsoft Analysis Services Projects".
- **Interfaz de Consulta:** SQL Server Management Studio (SSMS).

## 3. Modelo de Datos (Data Warehouse)

El núcleo de la solución es un **Data Warehouse** diseñado bajo la arquitectura de **Esquema de Estrella (Star Schema)**. Este modelo es el estándar en Business Intelligence porque optimiza la velocidad de lectura y simplifica la escritura de consultas analíticas al reducir el número de JOINS necesarios en comparación con un modelo relacional normalizado (3NF).

### 3.1 Tabla de Hechos (fact\_ventas)

Esta tabla representa el centro del universo del análisis y almacena los eventos transaccionales del negocio (las ventas). Su función es conectar las dimensiones con las métricas cuantitativas.

- **Granularidad:** Cada fila representa una venta individual ocurrida en un momento específico. Esta granularidad fina permite la máxima flexibilidad en el análisis.
- **Claves Foráneas (Foreign Keys):**
  - `producto_id`: Conecta con la dimensión de productos.
  - `pais_id`: Conecta con la dimensión geográfica.
- **Métricas (Measures):** Son los valores numéricos que serán objeto de operaciones matemáticas (suma, promedio, min, max).
  - `cantidad` (INT): Métrica aditiva que representa el volumen físico de productos movidos.
  - `total_dinero` (DECIMAL): Métrica aditiva que representa el ingreso económico. Al ser totalmente aditiva, se puede sumar coherentemente a través de todas las dimensiones (por país, por categoría, por fecha, etc.).

### 3.2 Tablas de Dimensión

Las dimensiones proporcionan el "contexto" a los hechos numéricos (el *Qué* y el *Dónde*). A diferencia de las bases de datos transaccionales, aquí las tablas están desnormalizadas intencionalmente para mejorar el rendimiento de lectura.

- **dim\_producto (Dimensión de Producto):**
  - Describe *qué* se vendió.
  - **Jerarquía de Catálogo:** Categoría -> Nombre.
  - Esta estructura permite operaciones de **Slicing** (Filtrado) para comparar el rendimiento entre líneas de negocio, respondiendo preguntas como: *"¿Qué categoría genera más ingresos, Electrónica o Ropa?"*.
- **dim\_pais (Dimensión Geográfica):**
  - Describe *dónde* ocurrió la venta.
  - **Jerarquía Geográfica:** Región -> País.

- Esta jerarquía es fundamental para las operaciones de **Drill-Down** (profundizar de Región a País) y **Roll-Up** (resumir de País a Región), permitiendo análisis a nivel continental o local.

## 4. Instrucciones de Instalación y Configuración

Para replicar este entorno de desarrollo, es necesario seguir estrictamente estas versiones y configuraciones para evitar errores de compatibilidad conocidos.

**Paso 1: Instalación del Motor de Base de Datos** Descargar e instalar **SQL Server 2019 Developer Edition**.

- **Nota importante:** Se optó por la versión "Developer Edition" (estable) en lugar de la versión 2025 (Preview) para evitar conflictos de librerías y errores de conexión con Visual Studio.
- Durante la instalación, en la sección "Feature Selection", marcar obligatoriamente:
  - Database Engine Services
  - Analysis Services

**Paso 2: Configuración de Analysis Services** En la pantalla de configuración de Analysis Services, seleccionar el modo **"Multidimensional and Data Mining Mode"**.

- *Advertencia:* Si se deja el modo "Tabular" por defecto, no se podrán crear cubos multidimensionales tradicionales.

**Paso 3: Preparación del Entorno de Diseño** Instalar **Visual Studio 2019 Community**.

1. En el instalador, seleccionar la carga de trabajo "Procesamiento de datos".
2. Una vez instalado, ir al menú Extensiones y descargar "Microsoft Analysis Services Projects".

## 5. Instrucciones para la Implementación de la Base de Datos

Utilizar SQL Server Management Studio (SSMS) para preparar el Data Warehouse.

**Paso 1: Conexión Inicial** Abrir SSMS y conectarse al servidor localhost (o .) utilizando autenticación de Windows.

**Paso 2: Ejecución del Script DDL** que se encuentra en el repositorio. Abrir una nueva consulta y ejecutar el código para crear la estructura.

**Paso 3: Carga de Datos (ETL Simulado)** Para tener datos suficientes para el análisis, ejecutar el siguiente bloque de código que simula 1,000 transacciones:

```
-- Generación automática de 1,000 ventas
DECLARE @i INT = 0;
WHILE @i < 1000
BEGIN
    INSERT INTO fact_ventas (producto_id, pais_id, cantidad, total_dinero)
    VALUES (
        FLOOR(RAND()*4+1),    -- Producto aleatorio
        FLOOR(RAND()*3+1),    -- País aleatorio
        FLOOR(RAND()*10+1),   -- Cantidad variable
        CAST((RAND()*500+10) AS DECIMAL(10,2)) -- Monto variable
    );
    SET @i = @i + 1;
END
```

## 6. Instrucciones para la Creación del Proyecto en Visual Studio

**Paso 1: Crear la Solución** Abrir Visual Studio y crear un nuevo proyecto de tipo "Analysis Services Multidimensional".

**Paso 2: Establecer el Origen de Datos (Data Source)**

1. Clic derecho en "Orígenes de datos" -> Nuevo origen de datos.
2. Conectar al servidor localhost y seleccionar la base OLAP\_Ventas\_DB.
3. **Configuración de Driver:** Asegurarse de seleccionar Microsoft OLE DB Driver for SQL Server en la lista de proveedores para garantizar estabilidad.

**Paso 3: Definir la Vista (Data Source View)** Crear una nueva vista seleccionando las tres tablas clave: fact\_ventas, dim\_producto y dim\_pais. Esto permite visualizar el esquema de estrella gráficamente.

**Paso 4: Diseño Asistido del Cubo**

1. Clic derecho en "Cubos" -> Nuevo Cubo.
2. Usar la opción "Usar tablas existentes".
3. Seleccionar fact\_ventas como la tabla de Grupo de Medidas.
4. Permitir que el asistente detecte las dimensiones automáticamente.

**Paso 5: Ajuste Manual de Atributos** Por defecto, el asistente solo activa los IDs. Para ver nombres reales:

1. Abrir la dimensión Dim Producto.
2. Arrastrar la columna Nombre desde la vista de origen a la lista de atributos.
3. Repetir el proceso para Dim País con las columnas País y Región.

## 7. Configuración Crítica de Seguridad y Permisos

Debido a los protocolos estrictos de SQL Server 2019, es necesario realizar ajustes manuales para permitir que el cubo procese los datos.

**Paso 1: La "Invitación VIP" (Permisos del Servicio)** El servicio de Analysis Services (MSSQLServerOLAPService) actúa como un usuario independiente. Para que pueda leer la base de datos relacional, se debe ejecutar la siguiente consulta en SSMS para otorgarle permisos de administrador (sysadmin), este mismo se encuentra en el repositorio para su uso:

```
USE [master];
GO
-- Crear el login para el servicio OLAP
CREATE LOGIN [NT SERVICE\MSSQLServerOLAPService] FROM WINDOWS;
GO
-- Otorgarle la "Llave Maestra" (Permisos de Sysadmin)
ALTER SERVER ROLE [sysadmin] ADD MEMBER [NT SERVICE\MSSQLServerOLAPService];
GO
```

*Esta acción es fundamental; sin ella, el cubo reportará errores de "Acceso denegado" al intentar procesar.*

**Paso 2: Desactivar Bloqueo de Certificados** En la cadena de conexión del Data Source en Visual Studio (Botón "Editar" -> "Avanzadas"), cambiar la propiedad:

- TrustServerCertificate = **True**

**Paso 3: Configuración de Suplantación** En la pestaña "Información de suplantación" del Origen de Datos, seleccionar **"Utilizar la cuenta de servicio"**. Esto instruye al cubo a usar las credenciales que autorizamos en el Paso 1.

## 8. Consultas MDX

Se realizaron pruebas de explotación del cubo mediante consultas MDX (Multidimensional Expressions).

### Operación: Drill-Down (Desglose)

Tomamos la medida general y bajamos un nivel de detalle para ver su distribución geográfica.

```
SELECT
    {[Measures].[Total Dinero]} ON COLUMNS,
    {[Dim Pais].[Pais].CHILDREN} ON ROWS
FROM [OLAP_Ventas_DB]
```

### Operación: Pivot (Rotación)

Cruzamos dos dimensiones (País en columnas y Producto en filas) creando una matriz de análisis.

```
SELECT
    {[Dim Pais].[Pais].MEMBERS} ON COLUMNS,
    {[Dim Producto].[Nombre].MEMBERS} ON ROWS
FROM [OLAP_Ventas_DB]
WHERE ([Measures].[Total Dinero])
```

### Operación: Dice (Trocear/Dado)

Recortamos un sub-cubo específico aplicando filtros simultáneos en múltiples dimensiones.

```
SELECT
    {[Measures].[Cantidad], [Measures].[Total Dinero]} ON COLUMNS,
    {[Dim Producto].[Nombre].MEMBERS} ON ROWS
FROM [OLAP_Ventas_DB]
WHERE (
    [Dim Pais].[Pais].&[Mexico],
    [Dim Producto].[Categoria].&[Electronica]
```

)

## 9. Conclusiones

- **Estabilidad de Versiones:** Es vital trabajar con versiones estables (2019 Developer) en lugar de versiones Preview (2025) para garantizar compatibilidad total con las librerías cliente de Visual Studio.
- **Gestión de Seguridad:** En entornos modernos de SQL Server, la configuración correcta de certificados (TrustServerCertificate) y permisos de suplantación son los pasos más críticos para un despliegue exitoso.
- **Herramientas:** Visual Studio ofrece una ventaja significativa sobre otras herramientas open-source gracias a su detección automática de esquemas estrella, agilizando el desarrollo del modelo multidimensional.

## 10. Autores

- Cruz Guzmán Carlos Alberto
- De La Rosa Hernández Tania
- Delgadillo Diaz Damián
- González González Erick Emiliano
- González Hernández Judith
- Magaña Fierro Elka Natalia
- Sánchez Ixmatlahua Kathia Jazmín
- Soto Nieves Uriel