



Laboratorio 2003

MDX

- **Temas:**
 - **Introducción a MDX**
 - **Modelo de datos**
 - **Funciones**
 - **Ejemplos**



Introducción a MDX

- **MultiDimensional eXpressions (MDX) es un lenguaje de definición y consultas sobre el modelo multidimensional**
 - Permite realizar consultas y cálculos.
 - Forma parte de la especificación OLE DB para OLAP.
 - Mediante este lenguaje las aplicaciones cliente pueden comunicarse con los proveedores OLAP
 - En nuestro caso Analysis Services.



Introducción a MDX(2)

■ Comparando SQL y MDX

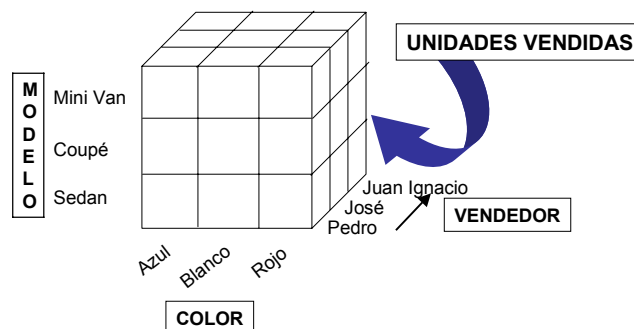
- Ambos son lenguajes de consulta y DDL
- Tienen similitudes en la sintaxis, pero diferencias en el significado.
 - Clausula SELECT
 - Clausula FROM
 - Clausula WHERE



Introducción a MDX (3)

■ EJEMPLO 1

- Trabajaremos sobre el siguiente cubo con información de ventas:





Introducción a MDX (3)

■ EJEMPLO 1

- Datos:

Modelo	Color	Vendedor	Unidades
Mini van	Azul	Juan Ignacio	2
Mini van	Azul	Pedro	2
Mini van	Azul	José	2
Coupé	Rojo	José	1
Sedan	Blanco	Juan Ignacio	3
Coupé	Blanco	José	2
Sedan	Rojo	Pedro	1
Sedan	Rojo	Juan Ignacio	2



Introducción a MDX (4)

- Se quiere obtener la cantidad de vehículos vendidos con modelo Sedan o Coupé, de color Rojo

<i>Color: Rojo</i>		Medida
		Unidades
Modelo	Coupe	1
	Sedan	3

```
SELECT
{Measures.Unidades} on columns,
{Modelo.Coupe,Modelo.Sedan} on rows
FROM CuboVentas
WHERE (Color.Rojo)
```

Introducción a MDX (5)

```
SELECT
{Measures.Unidades} on columns,
{Modelo.Coupe,Modelo.Sedan} on rows
FROM CuboVentas
WHERE (Color.Rojo)
```

Diagram illustrating the components of an MDX query:

- Especificación del eje 0**: Points to the `SELECT` clause.
- Especificación del eje 1**: Points to the `on columns` clause.
- Alcance de la consulta**: Points to the `FROM CuboVentas` clause.
- Restringe los resultados obtenidos a cierta rebanada del cubo. (opcional)**: Points to the `WHERE (Color.Rojo)` clause.

Facultad de Ingeniería - In.Co. *Sistemas de Data Warehousing* 7

Modelo de datos

- **Las expresiones en MDX se basan en el modelo de datos multidimensional:**
 - Cubos
 - Dimensiones y jerarquías
 - Niveles
- **Se utilizan los conceptos de :**
 - Miembros
 - Medidas
 - Tuplas
 - Conjuntos

Facultad de Ingeniería - In.Co. *Sistemas de Data Warehousing* 8



Modelo de datos (2)

■ Miembros:

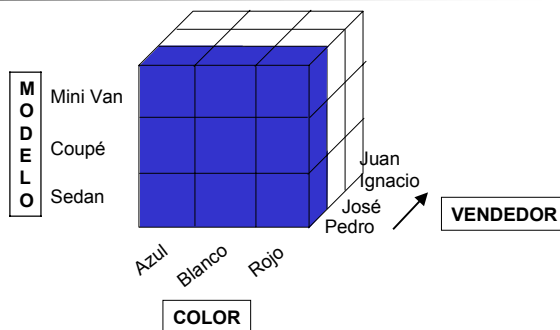
- Cada uno de los componentes de una dimensión.
- Representa el nivel mas bajo de granularidad.
- Los miembros pueden ser identificados:
 - Por su nombre
 - Por su clave
- También pueden definirse *miembros calculados*.
- Existen funciones para manejar miembros:
 - .Members
 - FirstChild
 - etc



Modelo de datos (3)

■ Miembros:

EJEMPLO:



Vendedor . Pedro



Modelo de datos (4)

■ Medida:

- Valores o indicadores a analizar
- Se organizan, por defecto, en una dimensión: Measures
- A partir de ellas pueden definirse *miembros calculados*, utilizando expresiones en MDX.



Modelo de datos (6)

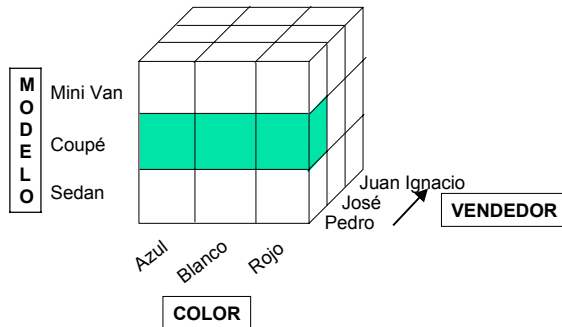
■ Tuplas:

- Combinación de miembros de diferentes dimensiones en la que puede incluirse, a lo sumo, un miembro de cada dimensión.
- Define una *rebanada del cubo* (slice), que eventualmente puede reducirse a una única celda.
- Se construye como utilizando () y separando los diferentes miembros que la componen con (,)
- **DIMENSIONALIDAD** de una tupla:
 - concepto que hace referencia a qué dimensiones componen la tupla y en qué orden lo hacen.



Modelo de datos (7)

■ Tuplas: EJEMPLOS:



(Modelo . Coupe , Vendedor . Pedro)



(Modelo . Coupe , Vendedor . Pedro , Vendedor . [Juan
Pedro])



Modelo de datos (8)

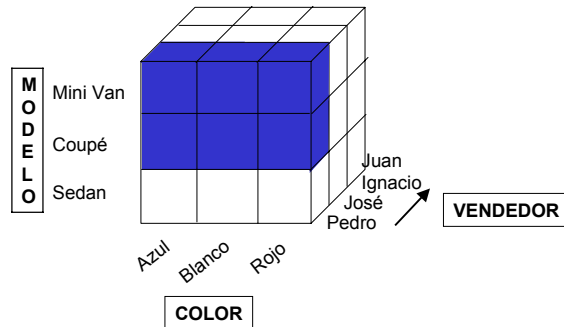
■ Conjuntos:

- Colección ordenada de tuplas .
- Todas las tuplas de un conjunto deben tener la misma dimensionalidad.
 - O sea:
 - Mismas dimensiones.
 - Mismo orden.
- Se construyen usando {}
- Son utilizados para:
 - Especificar ejes en la clausula SELECT
 - Restringir resultados (clausula WHERE)



Modelo de datos (9)

■ Conjunto: EJEMPLOS:



{ (Modelo . Coupé , Vendedor . Pedro) ,
(Modelo . [Mini Van] , Vendedor . Pedro) }



{ (Modelo . Coupé , Vendedor . Jose) ,
(Vendedor . Juan , Modelo . Coupé) }



Modelo de datos (10)

■ Consulta:

- Cada consulta en MDX se aplica sobre un único cubo.
- Los resultados obtenidos son transformaciones del cubo original.

■ Estructura básica

```
SELECT conjunto1 on columns,  
conjunto2 on rows,  
conjunto3 on pages,  
conjunto4 on chapters,  
conjunto5 on sections,
```

...

```
FROM cubo1  
WHERE conjunto
```




Operadores y Funciones

- **Operadores**
 - Corchetes { }
 - Parentesis rectos []
 - Parentesis curvos ()
 - Coma (,) y dos puntos (:)
- **Algunas funciones interesantes:**
 - .Members
 - CrossJoin()
 - Filter()
 - Order()



Operadores y Funciones (2)

- **Coma (,) y dos puntos (:)**
 - Mediante el uso de (,) se separan las tuplas que forman un conjunto.
 - En el caso de dimensiones en las que se definió un orden se puede utilizar (:) para indicar rangos.



Operadores y Funciones (3)

- **Coma (,) y dos puntos (:)**

EJEMPLOS:

1. `{Tiempo.[Enero - 2003] :Tiempo.[Julio - 2003]}`

 Crea un conjunto con los primeros seis meses del 2003.

2. `{Tiempo.[Enero - 2003], Tiempo.[Marzo - 2003]:Tiempo.[Julio - 2003]}`

 Crea un conjunto que contiene a enero y los meses comprendidos entre marzo y julio del 2003.



Operadores y Funciones (4)

- **.Members**

- Uno de los operadores MDX relacionados con recuperación de Metadata.
- Devuelve el conjunto de los miembros de la dimensión, jerarquía o nivel elegido.
- No se recuperan miembros calculados (ver función `AddCalculatedMembers()` [Spo01]).
- Existen otras funciones que permiten navegar las jerarquías:
 - Children
 - Siblings
 - FirstChild



Operadores y Funciones (5)

■ **.Members**

EJEMPLO:

1. `{Modelo.Members}`

Creará un conjunto con los miembros de la dimensión Modelo, (Mini Van, Coupé, Sedan).

El conjunto es análogo al siguiente:

```
{Modelo.[Mini Van], Modelo.Sedan, Modelo.Coupe}
```



Operadores y Funciones (6)

■ **CrossJoin()**

- Realiza el producto cartesiano entre dos conjuntos.
- En Analysis Services el operador (*) es un alias para este comando.



Operadores y Funciones (7)

■ CrossJoin()

EJEMPLO:

1. SELECT

```
CrossJoin({Modelo.Members}, {Color.Members}
```

```
) on columns ,
```

```
{Vendedor.Members} on rows
```

```
FROM CuboVentas
```

	Mini V	Mini V	Mini V	Coupe	Coupe	Coupe	Sedan	Sedan	Sedan
	azul	blanco	rojo	azul	blanco	rojo	azul	blanco	rojo
Juan	2	0	0	0	0	0	0	3	0
Jose	2	0	0	0	1	1	0	0	0
Pedro	2	0	0	0	0	0	0	0	1



Operadores y Funciones (7)

■ Filter()

- Selecciona un subconjunto del conjunto dado a partir de los criterios expresados mediante una expresion booleana .



Operadores y Funciones (8)

■ Filter()

EJEMPLOS:

1. Filter(
 {Modelo.Members},
 Measures.Unidades <=2)

Modelos que
vendieron 2 o
menos unidades

2. Filter(
 CrossJoin({Modelo.Members},
 {Color.Members}),
 Vendedor.Juan AND Measures.Unidades >3)

Parejas
(modelo,color) de
las cuales Juan
vendió mas de 3
unidades



Operadores y Funciones (9)

■ Order()

- Permite ordenar un conjunto según cierto criterio.
- Puede indicarse :
 - Orden ascendente (BASC)
 - Orden descente (BDESC)
 - Respetar orden jerárquico (ASC, DESC)



Operadores y Funciones (10)

■ Order()

EJEMPLO:

```
1. Order(  
    Filter(  
        CrossJoin({Modelo.Members},  
            {Color.Members}),  
        Vendedor.Juan AND  
        Measures.Unidades >3 ),  
    (Measures.Unidades), BDESC )
```



Parejas (modelo,color) de las cuales Juan vendió mas de 3 unidades, ordenadas en forma descendiente según la cantidad de unidades vendidas.



Material de referencia

- Nolan, C: "Introduction to Multidimensional Expressions (MDX)" <http://msdn.microsoft.com/library/en-s/dnolap/html/intromdx.asp> Microsoft Corporation, 1999
- MDX http://msdn.microsoft.com/library/default.asp?url=/library/en-us/olapdmad/agmdxbasics_90qg.asp, MSDN Library
- Reference guide, http://msdn.microsoft.com/library/default.asp?url=/library/en-us/oledb/html/olappr_chapter25_15.asp, MSDN Library



Bibliografía

- [Spo01] Spofford, G. :”MDX Solutions with Microsoft SQL Server Analysis Services”. ISBN number 0471400467. John Wiley & Son, Inc., 2001