



**ESCUELA POLITÉCNICA NACIONAL**  
**FACULTAD DE INGENIERÍA DE SISTEMAS**  
**INGENIERÍA DE SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN**

---

**Laboratorio de:** Inteligencia de negocios

**Práctica No.: 1**

**Tema: Operaciones multidimensionales**

**Nombre:** Danny Sebastián Díaz Padilla

**Fecha:** 30/10/2019

**1. Objetivos:**

**1.1. Objetivo General**

- Realizar las operaciones OLAP sobre el cubo de ventas de adventureWorksDW.

**1.2. Objetivos Específicos**

- Utilizar las operaciones: slice, dice, drilldown, rollup y pivote en SQL.
- Definir los operadores.
- Usar las consultas Cube y Roll Up propias de SQL

**2. Marco teórico:**

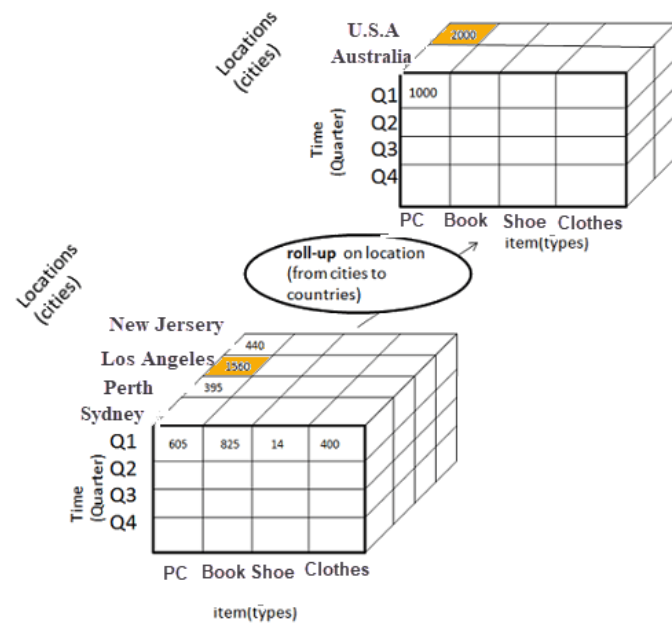
**Cubo OLAP**

El cubo OLAP es una estructura de datos optimizada para un análisis de datos muy rápido. Consiste en hechos numéricos llamados medidas que se clasifican por dimensiones. [1]

**Roll-up**

El roll-up también se conoce como "consolidación" o "agregación". La operación de roll-up se puede realizar de 2 maneras

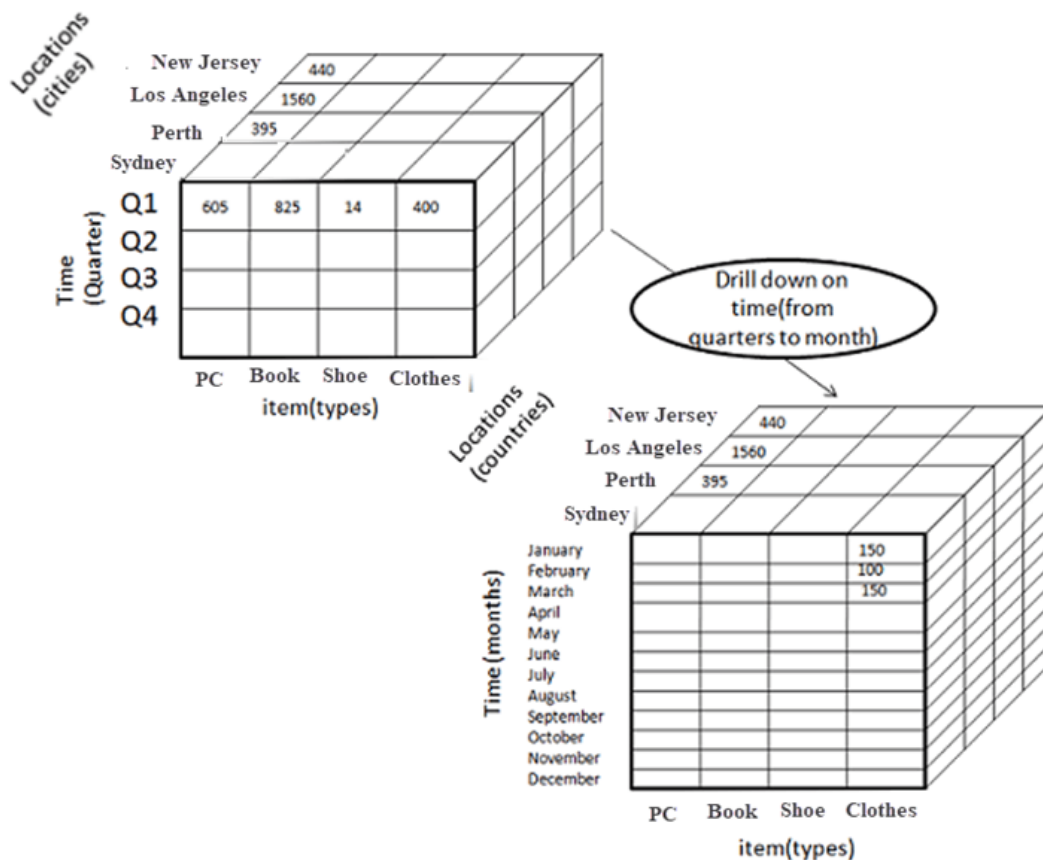
1. Dimensiones reducidas
2. Subiendo la jerarquía del concepto. La jerarquía conceptual es un sistema de agrupación de cosas en función de su orden o nivel.



## Drill Down

En el desglose, los datos se fragmentan en partes más pequeñas. Es lo opuesto al proceso de acumulación. Se puede hacer a través de

- Descendiendo por la jerarquía conceptual
- Aumentando una dimensión





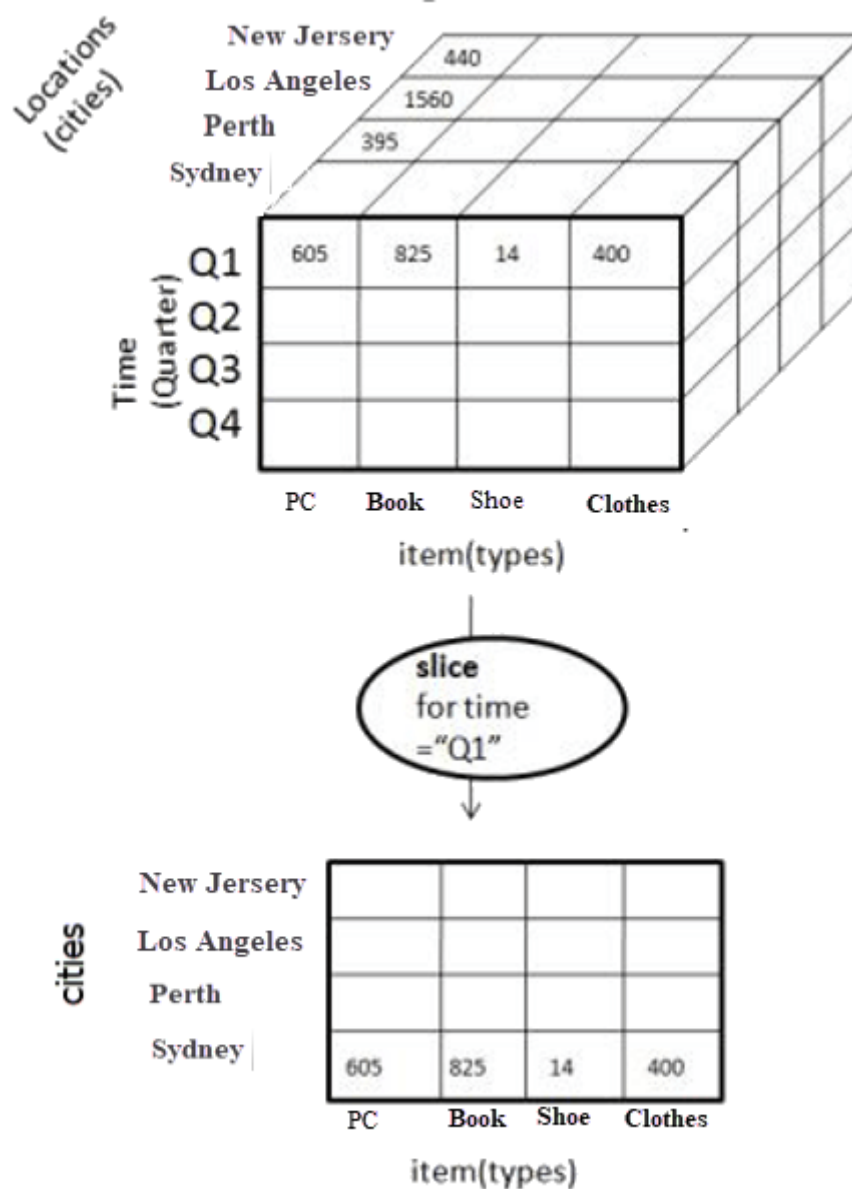
**ESCUELA POLITÉCNICA NACIONAL**  
**FACULTAD DE INGENIERÍA DE SISTEMAS**  
**INGENIERÍA DE SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN**

---

## Slice

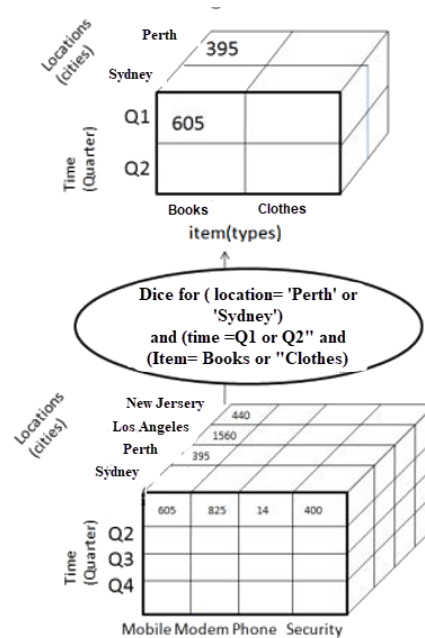
Aquí, se selecciona una dimensión y se crea un nuevo sub-cubo.

El siguiente diagrama explica cómo se realizó la operación de corte:



## Dice

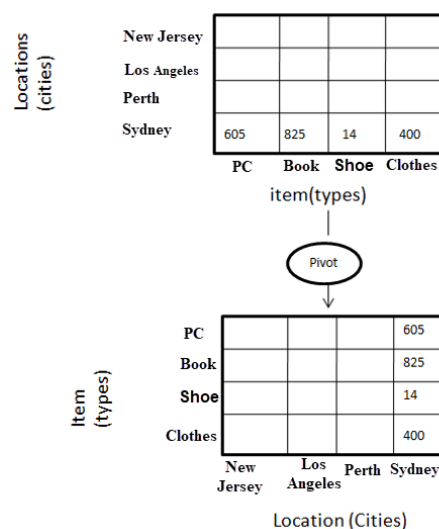
Esta operación es similar a una rebanada. La diferencia en Dice es que selecciona 2 o más dimensiones que dan como resultado la creación de un subcubo.



## Pivot

En Pivot, gira los ejes de datos para proporcionar una presentación sustituta de datos.

En el siguiente ejemplo, el pivote se basa en los tipos de elementos.





**ESCUELA POLITÉCNICA NACIONAL**  
**FACULTAD DE INGENIERÍA DE SISTEMAS**  
**INGENIERÍA DE SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN**

---

### 3. Desarrollo de la práctica:

use AdventureWorksDW2017;

La table de hechos a analizar tiene el siguiente aspecto:

```
SELECT * from dbo.FactInternetSales;
```

	ProductKey	OrderDateKey	DueDateKey	ShipDateKey	CustomerKey	PromotionKey	CurrencyKey	SalesTerritoryKey	SalesOrderNumber	SalesOrderLineNumber	Revisio
1	310	20101229	20110110	20110105	21768	1	19	6	SO43697	1	1
2	346	20101229	20110110	20110105	28389	1	39	7	SO43698	1	1
3	346	20101229	20110110	20110105	25863	1	100	1	SO43699	1	1
4	336	20101229	20110110	20110105	14501	1	100	4	SO43700	1	1
5	346	20101229	20110110	20110105	11003	1	6	9	SO43701	1	1
6	311	20101230	20110111	20110106	27645	1	100	4	SO43702	1	1
7	310	20101230	20110111	20110106	16624	1	6	9	SO43703	1	1
8	351	20101230	20110111	20110106	11005	1	6	9	SO43704	1	1
9	344	20101230	20110111	20110106	11011	1	6	9	SO43705	1	1
10	312	20101231	20110112	20110107	27621	1	100	4	SO43706	1	1

### Roll-up

Se obtiene el nombre de los clientes, el producto y el PAÍS.

```
SELECT dc.FirstName, dc.LastName, dp.EnglishProductName,  
dst.SalesTerritoryCountry, sum(fis.TotalProductCost) as SumaCostoTotal,  
fis.CustomerKey  
from dbo.FactInternetSales as fis, dbo.DimCustomer as dc, dbo.DimProduct as dp,  
dbo.DimSalesTerritory as dst  
where dc.CustomerKey = fis.CustomerKey and dp.ProductKey = fis.ProductKey and  
dst.SalesTerritoryKey = fis.SalesTerritoryKey  
group by dp.EnglishProductName, dc.FirstName, dc.LastName,  
dst.SalesTerritoryCountry, fis.SalesTerritoryKey, fis.CustomerKey;
```

	FirstName	LastName	EnglishProductName	SalesTerritoryCountry	SumaCostoTotal	CustomerKey
1	Andrea	Morris	All-Purpose Bike Stand	United States	59,466	11197
2	Kaitlyn	Hall	All-Purpose Bike Stand	United States	59,466	11682
3	Olivia	Peterson	All-Purpose Bike Stand	United States	59,466	11862
4	Alexandria	Rogers	All-Purpose Bike Stand	United States	59,466	12161
5	Jennifer	Campbell	All-Purpose Bike Stand	United States	59,466	12958
6	Lauren	Cox	All-Purpose Bike Stand	United States	59,466	13337
7	Ricardo	Shen	All-Purpose Bike Stand	United States	59,466	14905
8	Gabrielle	Collins	All-Purpose Bike Stand	United States	59,466	15975
9	Joe	Martinez	All-Purpose Bike Stand	United States	59,466	16004
10	Caleb	Evans	All-Purpose Bike Stand	United States	59,466	16072
11	Fernando	Thomas	All-Purpose Bike Stand	United States	59,466	23671

## Drill-down

Se agrega un nivel de detalle, en este caso el color del producto

```
SELECT dp.EnglishProductName, dst.SalesTerritoryCountry, dp.Color,
sum(fis.TotalProductCost) as SumaCostoTotal, fis.CustomerKey
from dbo.FactInternetSales as fis, dbo.DimCustomer as dc,  dbo.DimProduct as dp,
dbo.DimSalesTerritory as dst
where dc.CustomerKey = fis.CustomerKey and dp.ProductKey = fis.ProductKey and
dst.SalesTerritoryKey = fis.SalesTerritoryKey
group by dp.EnglishProductName, dc.FirstName, dc.LastName,
dst.SalesTerritoryCountry, dp.Color, fis.SalesTerritoryKey, fis.CustomerKey;
```

	EnglishProductName	SalesTerritoryCountry	Color	SumaCostoTotal	CustomerKey
2...	Mountain-200 Silver, ...	United States	Silver	1265,6195	12921
2...	Mountain-200 Silver, ...	United States	Silver	1265,6195	13220
2...	Mountain-200 Silver, ...	United States	Silver	1265,6195	13246
2...	Mountain-200 Silver, ...	United States	Silver	1265,6195	13455
2...	Mountain-200 Silver, ...	United States	Silver	1265,6195	13866
2...	Mountain-200 Silver, ...	United States	Silver	1265,6195	14374
2...	Mountain-200 Silver, ...	United States	Silver	1265,6195	14451
2...	Mountain-200 Silver, ...	United States	Silver	1265,6195	14464
2...	Mountain-200 Silver, ...	United States	Silver	1265,6195	15328
2...	Mountain-200 Silver, ...	United States	Silver	1265,6195	15399
2...	Mountain-200 Silver, ...	United States	Silver	1265,6195	15521

## Pivot

Solo cambia el orden de las columnas.

```
SELECT dp.EnglishProductName, dc.FirstName, dc.LastName,
dst.SalesTerritoryCountry, dp.Color, sum(fis.TotalProductCost) as SumaCostoTotal,
fis.CustomerKey
from dbo.FactInternetSales as fis, dbo.DimCustomer as dc,  dbo.DimProduct as dp,
dbo.DimSalesTerritory as dst
where dc.CustomerKey = fis.CustomerKey and dp.ProductKey = fis.ProductKey and
dst.SalesTerritoryKey = fis.SalesTerritoryKey
group by dp.EnglishProductName, dc.FirstName, dc.LastName,
dst.SalesTerritoryCountry, dp.Color, fis.SalesTerritoryKey, fis.CustomerKey;
```

	EnglishProductName	FirstName	LastName	SalesTerritoryCountry	Color	SumaCostoTotal	CustomerKey
1	All-Purpose Bike Stand	Jasmine	Taylor	United States	NA	59,466	11137
2	All-Purpose Bike Stand	Rebecca	Young	United States	NA	59,466	14418
3	All-Purpose Bike Stand	Rachel	Lee	United States	NA	59,466	14575
4	All-Purpose Bike Stand	Hailey	Morgan	United States	NA	59,466	22753
5	All-Purpose Bike Stand	Jodi	Becker	United States	NA	59,466	26043
6	All-Purpose Bike Stand	Jonathan	Patterson	United States	NA	59,466	29034
7	All-Purpose Bike Stand	Chloe	Young	United States	NA	59,466	11015
8	All-Purpose Bike Stand	Tamara	Chander	United States	NA	59,466	11741
9	All-Purpose Bike Stand	Kevin	Butler	United States	NA	59,466	14450
10	All-Purpose Bike Stand	Jerry	Ferrier	United States	NA	59,466	14730
11	All-Purpose Bike Stand	Samuel	Ross	United States	NA	59,466	15957



**ESCUELA POLITÉCNICA NACIONAL**  
**FACULTAD DE INGENIERÍA DE SISTEMAS**  
**INGENIERÍA DE SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN**

---

**Slice**

Se parte de acuerdo al producto y a la promoción:

```
SELECT dp.EnglishProductName, dpr.EnglishPromotionName, fis.SalesAmount,  
dpr.DiscountPct  
from dbo.FactInternetSales as fis, dbo.DimProduct as dp, dbo.DimPromotion as dpr  
where dp.ProductKey = fis.ProductKey and dpr.PromotionKey = fis.PromotionKey;
```

	EnglishProductName	EnglishPromotionName	SalesAmount	DiscountPct
1	Road-150 Red, 62	No Discount	3578,27	0
2	Mountain-100 Silver, 44	No Discount	3399,99	0
3	Mountain-100 Silver, 44	No Discount	3399,99	0
4	Road-650 Black, 62	No Discount	699,0982	0
5	Mountain-100 Silver, 44	No Discount	3399,99	0
6	Road-150 Red, 44	No Discount	3578,27	0
7	Road-150 Red, 62	No Discount	3578,27	0
8	Mountain-100 Black, 48	No Discount	3374,99	0
9	Mountain-100 Silver, 38	No Discount	3399,99	0
10	Road-150 Red, 48	No Discount	3578,27	0
11	Road-150 Red, 48	No Discount	3578,27	0

**Cube (SQL Server)**

Función de SQL que agrupa como un cubo a los datos

```
SELECT dp.EnglishProductName, dst.SalesTerritoryCountry, sum(fis.SalesAmount) as  
SumaVentasTotal  
from dbo.FactInternetSales as fis, dbo.DimCustomer as dc, dbo.DimProduct as dp,  
dbo.DimSalesTerritory as dst  
where dc.CustomerKey = fis.CustomerKey and dp.ProductKey = fis.ProductKey and  
dst.SalesTerritoryKey = fis.SalesTerritoryKey  
group by CUBE(dp.EnglishProductName, dc.FirstName, dc.LastName,  
dst.SalesTerritoryCountry, fis.SalesTerritoryKey, fis.CustomerKey);
```

	EnglishProductName	SalesTerritoryCountry	SumaVentasTotal
1	Fender Set - Mountain	Australia	21,98
2	Mountain-100 Silver, 38	Australia	3399,99
3	Mountain-200 Silver, 38	Australia	2319,99
4	Short-Sleeve Classic Jersey, S	Australia	53,99
5	Sport-100 Helmet, Red	Australia	34,99
6	Touring Tire	Australia	28,99
7	Touring Tire Tube	Australia	4,99
8	Touring-1000 Blue, 46	Australia	2384,07
9	NULL	Australia	8248,99
10	NULL	Australia	8248,99
11	NULL	Australia	8248,99

## ROLL-UP (SQL Server)

Es la función que ya trae integrada SQL Server

```
SELECT dp.EnglishProductName, dst.SalesTerritoryCountry, sum(fis.SalesAmount) as SumaVentasTotal
from dbo.FactInternetSales as fis, dbo.DimCustomer as dc, dbo.DimProduct as dp,
dbo.DimSalesTerritory as dst
where dc.CustomerKey = fis.CustomerKey and dp.ProductKey = fis.ProductKey and
dst.SalesTerritoryKey = fis.SalesTerritoryKey
group by ROLLUP(dp.EnglishProductName, dc.FirstName, dc.LastName,
dst.SalesTerritoryCountry, fis.SalesTerritoryKey, fis.CustomerKey);
```

	EnglishProductName	SalesTerritoryCountry	SumaVentasTotal
1	All-Purpose Bike Stand	United States	159,00
2	All-Purpose Bike Stand	United States	159,00
3	All-Purpose Bike Stand	United States	159,00
4	All-Purpose Bike Stand	NULL	159,00
5	All-Purpose Bike Stand	NULL	159,00
6	All-Purpose Bike Stand	United States	159,00
7	All-Purpose Bike Stand	United States	159,00
8	All-Purpose Bike Stand	United States	159,00
9	All-Purpose Bike Stand	NULL	159,00
10	All-Purpose Bike Stand	United Kingdom	159,00
11	All-Purpose Bike Stand	United Kingdom	159,00





**ESCUELA POLITÉCNICA NACIONAL**  
**FACULTAD DE INGENIERÍA DE SISTEMAS**  
**INGENIERÍA DE SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN**

---

**4. Análisis de resultados:**

Se usó una versión del 2017 de la base de datos de Data warehouse debido a su disponibilidad y compatibilidad con sistemas y entornos cercanos.

Utilizar varios join para obtener la juntura que necesitamos, sobrecarga el desempeño.

**5. Conclusiones y recomendaciones:**

- OLAP pre-procesa la información y nos da ventajas en futuros análisis.
  - SQL Server ya trae integrado algunas funciones para el cubo de ventas.
  - Es necesario agrupar los datos cuando realizamos operaciones sobre toda una columna.
  - Se logró utilizar las operaciones: slice, dice, drilldown, rollup y pivote en SQL.
  - Los operadores quedaron definidos de una forma más clara.
  - Se logró usar las consultas Cube y Roll Up propias de SQL
- 
- Se recomienda no considerar datos nulos para evitar pérdidas de calidad en futuros análisis de datos.
  - Es recomendable verificar la granularidad con la que trabaja la tabla o las tablas de hechos.

**6. Bibliografía:**

- [1] «Guru99,» [En línea]. Available: <https://www.guru99.com/online-analytical-processing.html>. [Último acceso: 30 Octubre 2019].