

بسم الله الرحمن الرحيم
جامعة الخرطوم
كلية العلوم الرياضية - المستوى الخامس (حاسوب)
قواعد البيانات المتقدمة

مستودعات البيانات Data Warehousing

أداء الطلاب:

- | | |
|-------|---------------------------|
| ٩٩-٦٢ | ١- وليد محمد الحسن عمر |
| ٩٩-٢٤ | ٢- عبدالباسط عبدالصمد آدم |

قبل المضي قدماً يجب تعريف المصطلحات التالية:

- * **OLAP (On-line Analytical Processing)** : هي عملية تحليل البيانات المتراكمة في مستودع البيانات.
- **(Decision Support Systems) DSS**: وتعرف كذلك بـ **EIS (Executive Information Systems)** وهي أنظمة تساعد القياديين في المؤسسات والمنظمات على اتخاذ القرارات الحاسمة والمعقدة وذلك بتوفير بيانات من مستوى عالي.
- **تنقيب البيانات (Data Mining)**: هي عملية البحث داخل مستودع البيانات عن معرفة غير مستكشفة (اكتشاف المعرفة).
- **(On-line Transaction Processing) OLTP**: هي مجموعة عمليات الإضافة والحذف والتعديل بالإضافة إلى الاستعلام مع بعض التحليل الذي لا يرقى لأن يكون مساعداً في اتخاذ القرارات. ويتوفر دعم هذه العمليات في قواعد البيانات التقليدية.

ماهي مستودعات البيانات؟

هي مجموعة من البيانات دائمة تاريخية متكاملة للمساعدة في اتخاذ القرارات الإدارية. فهي تساعد على الوصول للبيانات لأغراض التحليلات الزمنية واكتشاف المعرفة واتخاذ القرارات لأنها مصممة خصيصاً لاستخلاص البيانات ومعالجتها وتمثيلها وتقديمها بصورة مناسبة لهذه الأغراض، وتتضمن كمية ضخمة من البيانات قد تكون من مصادر مختلفة، مثلاً عدة قواعد بيانات من عدة نماذج بيانات، وأحياناً من أنظمة ومنصات مختلفة.

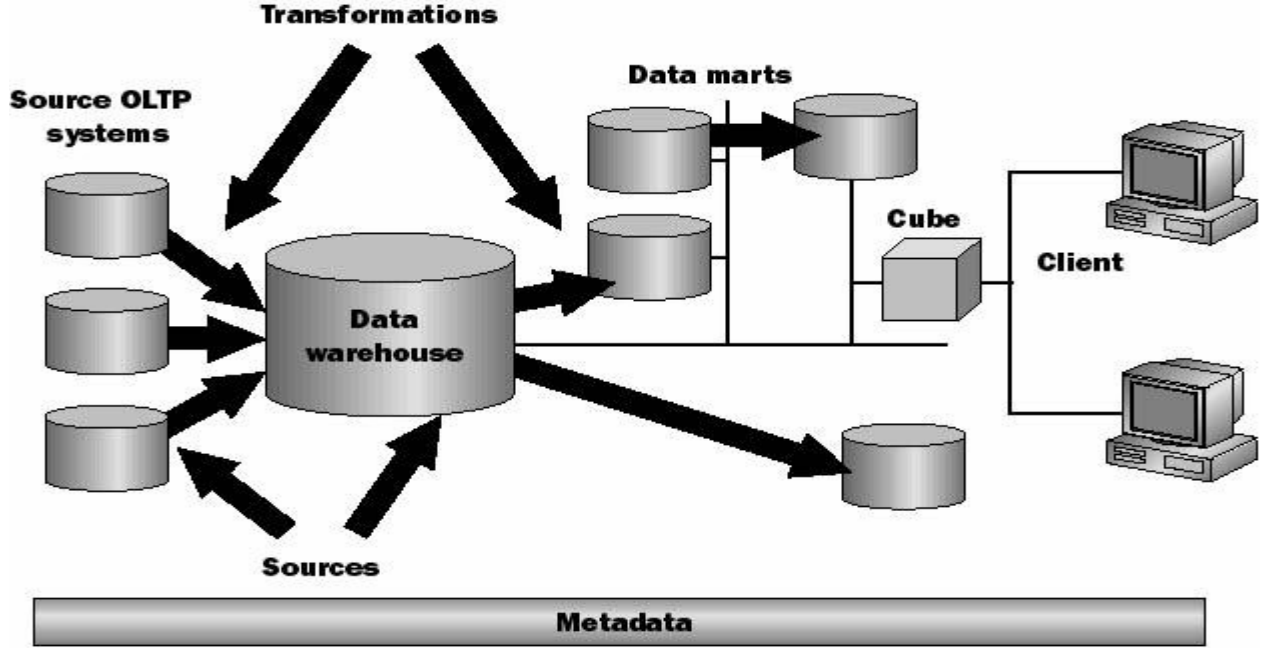
خصائص مستودعات البيانات:

- (١) تستخدم النموذج متعدد الأبعاد (Multidimensional Model).
- (٢) تدعم السلاسل الزمنية (Time Series) وتحليل التوجهات (Trends Analysis) اللذين يحتاجان لبيانات تاريخية لاتستطيع قواعد البيانات العادية (Transactional Databases) أن توفرها.
- (٣) تحديث البيانات فترى (Periodic) أي يتم كل فترة بواسطة أجزاء منه تختص بهذا الأمر.
- (٤) استرجاع البيانات وتحليلها هو صميم عملها، وتهتم به أكثر.
- (٥) مستويات تجميع (Aggregation) وأبعاد (Dimensions) غير محدودة.
- (٦) دعم معمارية Client/server وتعددية المستخدمين.
- (٧) الاحتفاظ بكمية ضخمة من البيانات قد تصل إلى عدة تيرابايتات (1 TB = 1024 GB).

الخاصية الأخيرة شكلت مشكلة، ولكن تم حلها بابتكار كل من الآتي:

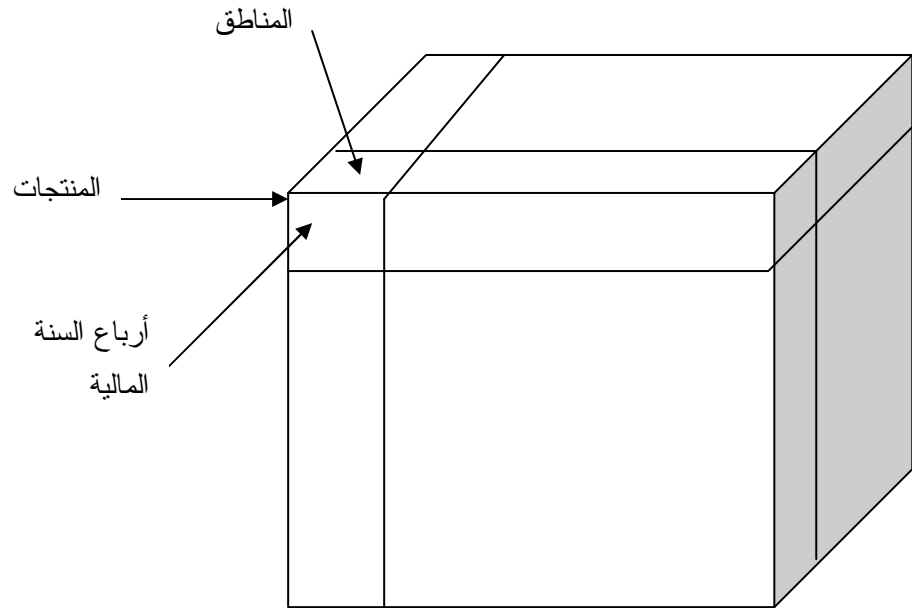
- ١- **مستودعات البيانات الشاملة (Enterprise-wide Data Warehouses)**: وهي مشاريع ضخمة تتطلب استثماراً ضخماً في الوقت والموارد.
- ٢- **مستودعات البيانات الافتراضية (Virtual Data Warehouses)**: وهي استعلامات على قواعد البيانات الوظيفية مصممة بكفاءة عالية للوصول السريع للبيانات.
- ٣- **متاجر البيانات (Data Marts)**: هي أجزاء من مستودع البيانات موجهة لجزء من المؤسسة (كقسم معين منها).

معمارية مستودعات البيانات:



نمذجة البيانات داخل مستودع البيانات:

كما ذكرنا، مستودعات البيانات تستخدم النموذج متعدد الأبعاد، وتستفيد من العلاقات بين البيانات لتسكينها في مصفوفات متعددة الأبعاد تسمى مكعبات البيانات (Data Cubes)، وتسمى بالمكعبات الفائقة (Hyper Cubes) إذا احتوت على أكثر من ثلاثة أبعاد. البيانات المخزنة في هذا النموذج أفضل من ناحية أداء الاستعلامات من مثيلاتها المخزنة في النموذج العلائقي. مثال للأبعاد: في مستودع بيانات خاص بشركة تجارية: أرباع السنة المالية، المنتجات، المناطق. وبإضافة بعد آخر (الزبائن مثلاً) يتحول لمكعب فائق، مع أنه صعب التخيل والتمثيل.



في هذا المثال كل خلية في المكعب تحتوي على بيانات بضاعة معينة في ربع سنة مالية معين في منطقة معينة. عملية الانتقال من تدرج بعد لآخر تسمى بالارتكاز (Pivoting) أو الدوران (Rotation)، مثلاً تدوير المكعب لعرض بيانات المناطق كصفوف ومجموع المبيعات في أرباع السنة المالية كأعمدة، وأنواع البضائع كبعد ثالث. لاحظ تشابهه مع دراسة مجسم دالة متعددة المتغيرات في فضاء متعدد الأبعاد.

تخزين النموذج متعدد الأبعاد يتضمن نوعين من الجداول:

١- **جدول البعد (Dimension Table):** وصفه تصف سمات attributes البعد.

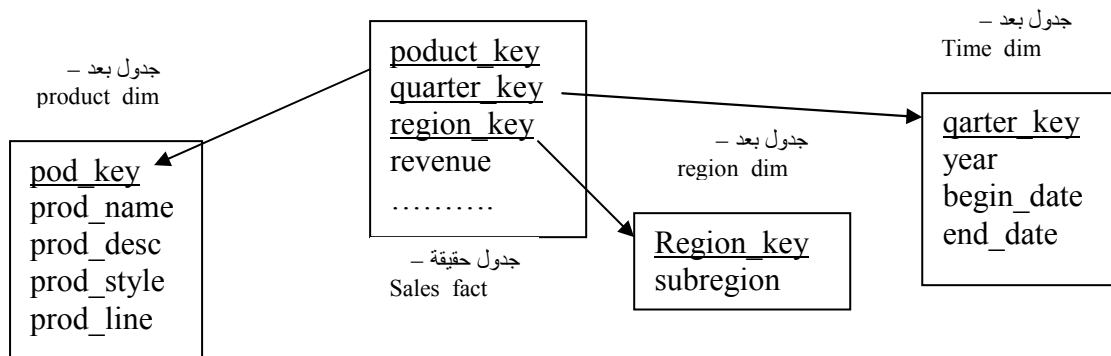
٢- **جدول الحقيقة (Fact Table):** ويتضمن مقاييس أداء العمل (Business Key Performance Indicators)، وتعرّف بمؤشرات لجدول الأبعاد، ويتضمن هذا الجدول البيانات.

توجد بنيتين شائعتين لتعريف هذه الجداول، هما:

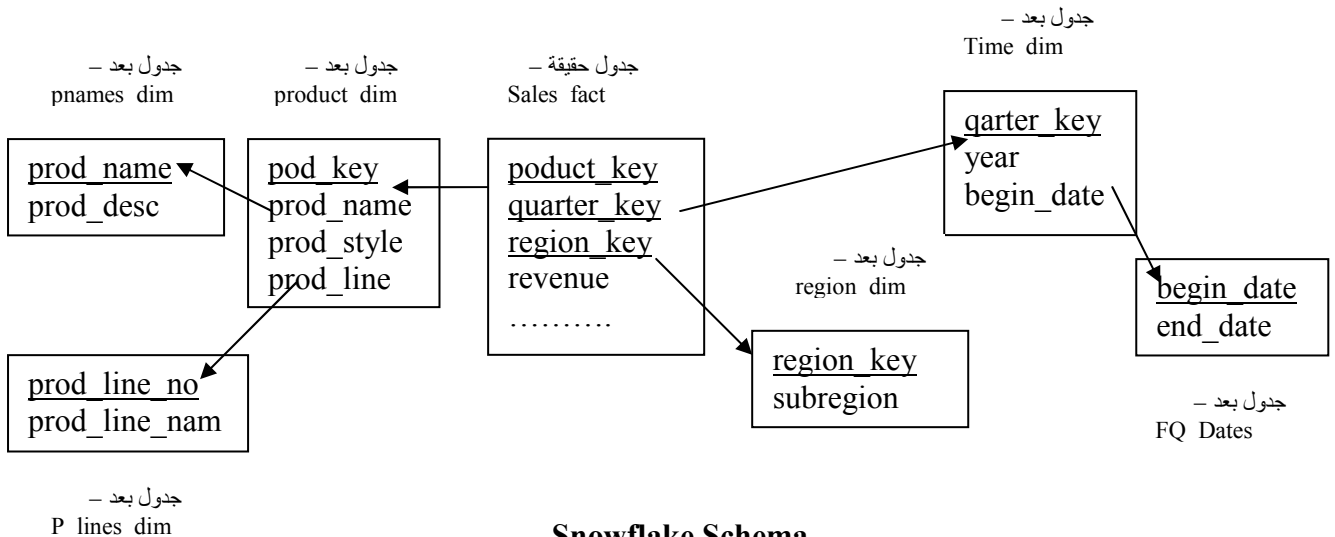
١/ **Star Schema:** وتتكون من جدول الحقيقة مع جدول وحيد لكل بعد. ويعيها احتياجها لمساحة كبيرة لتخزين البيانات.

٢/ **Snowflake Schema:** وهي تعديل للStar Schema وذلك بتطبيق قوانين التطبيع (Normalization) عليها. ويعيها تضييعها لطاقة الجهاز المضيف في عمليات الربط (Joining).

مثال:



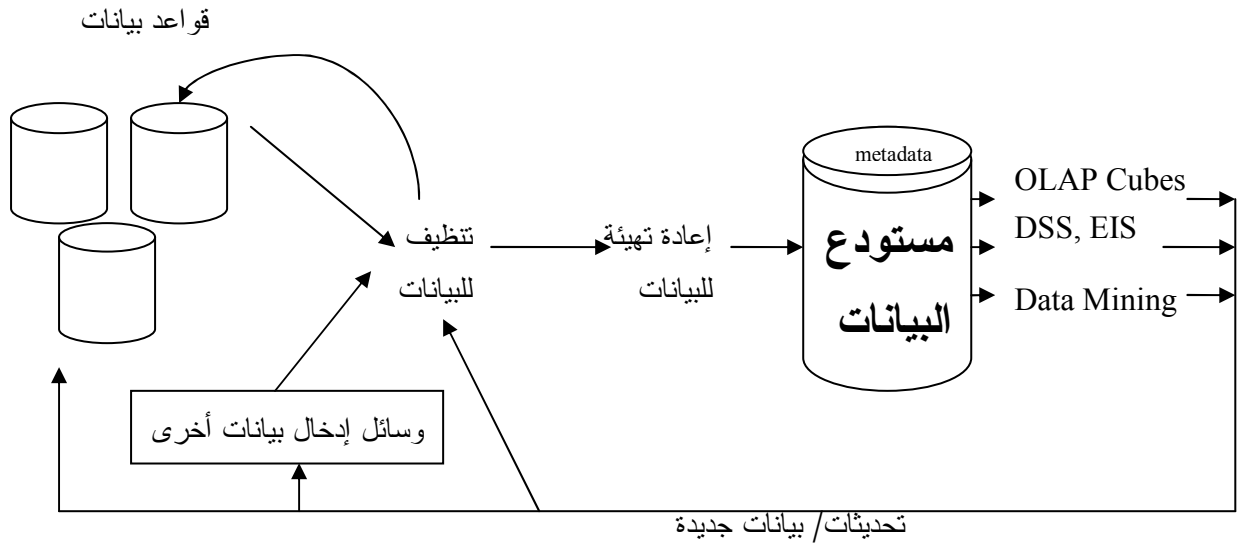
Star Schema



Snowflake Schema

بناء مستودعات البيانات:

العمليات التي تتم لإنجاز هذا العمل موضحة في الشكل التالي:



كما توجد بعض القضايا التي تجب مراعاتها عند بناء مستودعات البيانات ومنها:

- ١- استخلاص البيانات من عدة مصادر قد تكون غير متجانسة.
- ٢- تهيئة البيانات لضمان تلاؤمها (Consistency) داخل مستودع البيانات.
- ٣- تنظيف البيانات لضمان شرعيتها (Validity)، ويتم ذلك في قاعدة البيانات التي تم أخذ البيانات منها.
- ٤- مراقبة وضبط حجم مستودع البيانات خلال وبعد تحميله بالبيانات.
- ٥- كل كم فترة يجب تحديث البيانات فيه؟
- ٦- ماهو الزمن اللازم لبنائه؟ وماهي جدواه الاقتصادية بالنسبة للمؤسسة التي استعملته؟
- ٧- هل نحتاج لأن يكون موزعا (Distributed) أم مركزيا (Centralized)؟

الوظائف داخل مستودع البيانات:

- ١- Roll-up: حيث يتم تلخيص البيانات في شكل عمومي متزايد (أسبوعياً إلى ربعياً إلى سنوياً).
- ٢- Drill-Down: حيث تتم زيادة مستويات تفصيل البيانات، بعكس ال Roll-up.
- ٣- Pivot (Rotation): سبق شرحها أعلاه.
- ٤- Slice and Dice: تنفيذ عمليات الإسقاط على الأبعاد.
- ٥- الترتيب (Sorting): ترتيب البيانات بقيمة قابلة للترتيب.
- ٦- الاختيار (Selection): اختيار البيانات بقيمة أو مدى قيم.
- ٧- الصفات المحسوبة: وهي قيم لصفات يتم حسابها بعمليات على القيم المخزنة والمشتقة.

الفرق ما بين مستودعات البيانات والمرئيات: مع أنهما يبدوان متشابهين لأول وهلة، إلا أن بينهما الفروقات التالية:

- ١- مستودعات البيانات تتواجد كتخزين دائم ولا تتشكل عند الطلب كالمرئيات.
- ٢- مستودعات البيانات ليست دائماً علائقية، بل متعددة الأبعاد.
- ٣- مستودعات البيانات يمكن فهرستها لتحسن الأداء، بينما لا يمكن فهرسة المرئيات مستقلةً عن جداولها القاعدية.
- ٤- مستودعات البيانات تستطيع عمل الوظائف المذكورة أعلاه، بينما المرئيات لا تستطيع عملها كلها.
- ٥- مستودعات البيانات تعطي تخزيناً ضخماً لبيانات زمنية بصورة أكبر من تلك المحتواة في قاعدة البيانات، بينما المرئيات هي خلاصة قاعدة البيانات.

الأدوات: دوال (Relational OLAP) ROLAP، دوال (Multidimensional OLAP) MOLAP، امتدادات من SQL وطرق ربط متقدمة (Advanced Join Methods) ومسح ذكي (Intelligent Scanning)، ويتم تحسين أداء هذه الأدوات باستخدام المعالجة المتوازية.

الصعوبات التي تواجه تطبيق مستودعات البيانات:

- ١/ الإنشاء يستغرق بعض الزمن، ابتداءً من وضع الخطط حوله وحتى الانتهاء من تطبيقه.
 - ٢/ إدارة المستودع صعبة نظراً لـ كبر حجمه وتعقيده وتتطلب تدريباً أكثر للقائمين عليه، وخصوصاً من ناحية مراعاة جودة البيانات (Data quality).
 - ٣/ تقدير احتياجات مستخدميه قبل إنشائه.
 - ٤/ ظهور منابع جديدة للبيانات بعد الانتهاء من إنشائه يزيد من صعوبة إدارته.
-

مثال تطبيقي لمستودع بيانات في قاعدة البيانات SQL Server: فيما يلي جمل SQL التي تكون جداوله الأساسية (جداول الأبعاد و جدول الحقيقة):

```
create database Northwind_s_mart
go
use Northwind_s_mart
go
```

```
CREATE TABLE [dbo].[Customer_Dim] (
    [customer_key] [int] IDENTITY (1, 1) NOT NULL primary key,
    [customer_id] [nvarchar] (5) NOT NULL ,
    [region] [nvarchar] (15) NULL ,
)
GO
```

```
CREATE TABLE [dbo].[Product_Dim] (
    [product_key] [int] IDENTITY (1, 1) NOT NULL primary key,
    [product_id] [int] NOT NULL ,
    [product_name] [nvarchar] (40) NOT NULL ,
    [prod_supplier_name] [nvarchar] (40),
    [prod_category_name] [varchar] (15) NOT NULL ,
    [list_unit_price] [money] NOT NULL
)
GO
```

```
CREATE TABLE [dbo].[Time_Dim] (
    [time_key] [int] IDENTITY (1, 1) NOT NULL primary key,
    [the_date] [datetime] NOT NULL ,
    [day_of_week] [nvarchar] (50) NOT NULL ,
    [month] [int] NOT NULL ,
    [year] [int] NOT NULL ,
    [quarter] [int] NOT NULL ,
    [day_of_year] [int] NOT NULL ,
    [holiday] [nvarchar] (50) NOT NULL ,
    [weekend] [nvarchar] (50) NOT NULL ,
    [year_month] [nvarchar] (50) NOT NULL ,
    [week_of_year] [int] NOT NULL
)
```

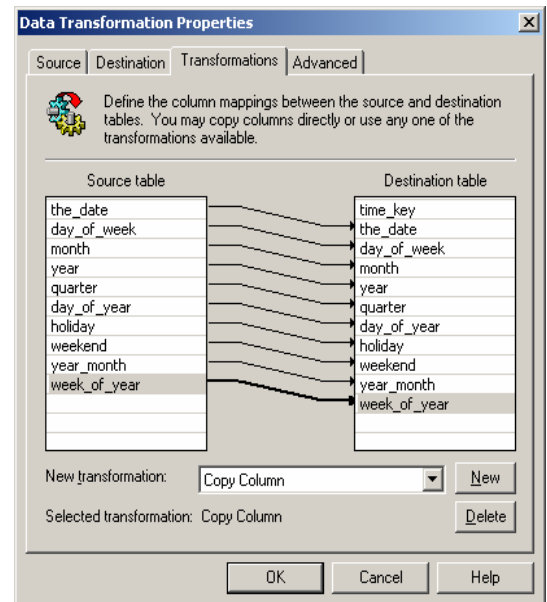
```
)
GO
```

```
CREATE TABLE [dbo].[Sales_Fact] (
    [time_key] [int] NOT NULL ,
    [customer_key] [int] NOT NULL ,
    [product_key] [int] NOT NULL ,
    [required_date] [datetime] NOT NULL ,
    [line_item_freight] [money] NOT NULL ,
    [line_item_total] [money] NOT NULL ,
    [line_item_quantity] [int] NOT NULL ,
    [line_item_discount] [real] NOT NULL,
    primary key(time_key, customer_key, product_key),
    foreign key(time_key) references Time_Dim(time_key),
    foreign key(customer_key) references Customer_Dim(customer_key),
    foreign key(product_key) references Product_Dim(product_key)
)
GO
```

```
select distinct s.ShippedDate as the_date,
    datename(dw, s.shippeddate) as day_of_week,
    datepart(mm, s.shippeddate) as [month],
    datepart(yy, s.shippeddate) as [year],
    datepart(qq, s.shippeddate) as [quarter],
    datepart(dy, s.shippeddate) as day_of_year,
    'N' as holiday,
    case datepart(dw, s.shippeddate)
        when (1) then 'Y'
        when (7) then 'Y'
        else 'N'
    end as weekend,
    datename(month, s.shippeddate)+'_'+
    datename(year,s.shippeddate) as year_month,
    datepart(wk, s.shippeddate) as week_of_year

from orders s
where s.shippeddate is not null
```

وجملة SQL التالية تنقل البيانات لجدول البعد الزمني:



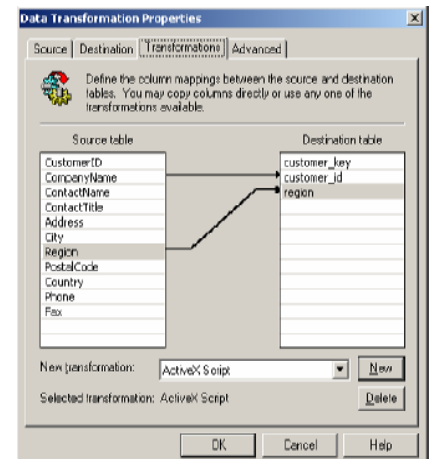
وبالنسبة لبعد الزبائن: الدالة التالية تعيد ملء الحقل region بالقيمة 'others' إذا كان null:

```
Function Main()
    if IsNull(DTSSource("Region")) then
        DTSDestination("Region")="Other"
    else
        DTSDestination("Region") = DTSSource("Region")
    end if
    Main = DTSTransformStat_OK
End Function
```

وتوضع في الـ activeX Script.

وبالنسبة لبعد البضائع: جملة SQL التالية تنقل البيانات لجدول بعد البضائع:

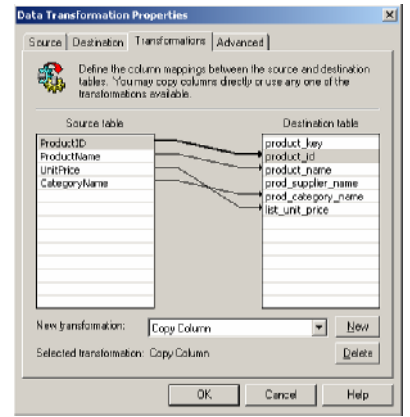
```
SELECT Products.ProductID, Products.ProductName,
```



Products.UnitPrice, Categories.CategoryName
 FROM Products INNER
 JOIN Categories ON
 Products.CategoryID = Categories.CategoryID

وأخيرا جملة SQL التي تملأ جدول الحقيقة (sales_fact):

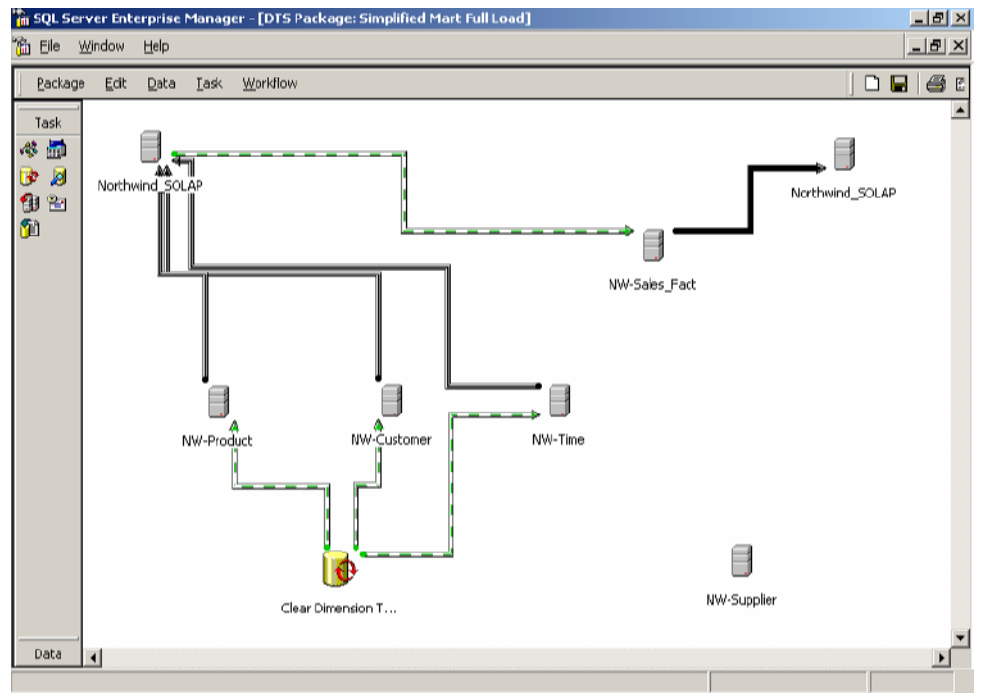
```
SELECT
    Northwind_Mart.dbo.Time_Dim.TimeKey,
    Northwind_Mart.dbo.Customer_Dim.CustomerKey,
    Northwind_Mart.dbo.Product_Dim.ProductKey,
    Northwind.dbo.Orders.Required_Date,
    Orders.Freight * [Order Details].Quantity /
        (SELECT SUM(Quantity)
         FROM [Order Details] od
         WHERE od.OrderID = Orders.OrderID) AS LineItemFreight,
    [Order Details].UnitPrice * [Order Details].Quantity AS LineItemTotal,
    [Order Details].Quantity AS LineItemQuantity,
    [Order Details].Discount * [Order Details].UnitPrice *
        [Order Details].Quantity AS LineItemDiscount
FROM Orders
INNER JOIN [Order Details]
    ON Orders.OrderID = [Order Details].OrderID
INNER JOIN Northwind_Mart.dbo.Product_Dim
    ON [Order Details].ProductID =
        Northwind_Mart.dbo.Product_Dim.Product_ID
INNER JOIN Northwind_Mart.dbo.Customer_Dim
    ON Orders.CustomerID =
        Northwind_Mart.dbo.Customer_Dim.Customer_ID
INNER JOIN Northwind_Mart.dbo.Time_Dim
    ON Orders.ShippedDate = Northwind_Mart.dbo.Time_Dim.The_Date
WHERE (Orders.ShippedDate IS NOT NULL)
```



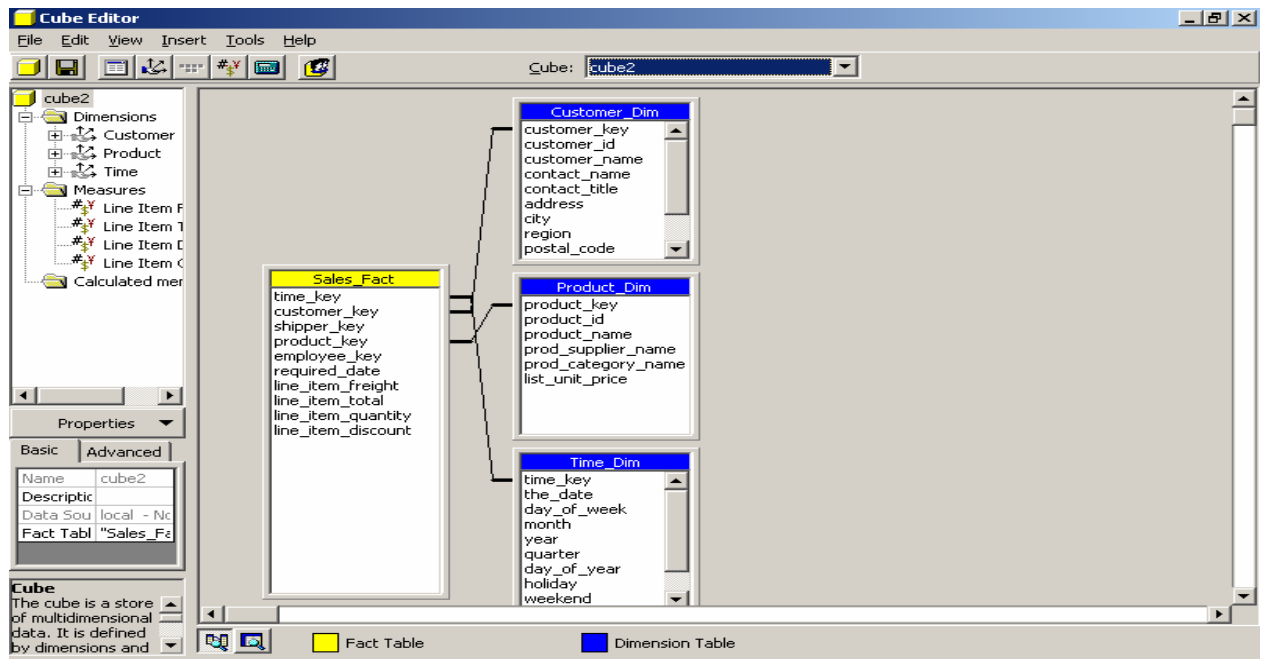
الشكل التالي يحدد خارطة انسياب البيانات من قاعدة البيانات التي نريد عمل مستودع للبيانات التي فيها، إلى مستودع البيانات، بالإضافة لتسلسل أداء المهام.

Package -> execute

تقوم بتنفيذ مهام نقل البيانات التي حددناها لها. وبعد الانتهاء من نقل البيانات نقوم بإنشاء مكعب البيانات من البرنامج OLAP Service Manager.



وهذا هو مكعب البيانات بعد الانتهاء من إنشائه، لاحظ أننا استخدمنا بنية Star Schema لتعريف الجداول:



نفس المكعب ولكن في وضع استعراض البيانات:

		MeasuresLevel		
		Line Item Freight	Line Item Total	Line Item Discount
- Country	+ Region			
- Austria	+ Other			
- Belgium	Belgium Total			
	+ Other			
- Brazil	Brazil Total	\$ 26.83	\$ 380.00	0.00
	+ RJ	\$ 26.83	\$ 380.00	0.00
	+ SP			
- Canada	Canada Total			
	+ BC			
	+ Québec			
- Denmark	Denmark Total			
	+ Other			
- Finland	Finland Total			
	+ Other			
- France	France Total	\$ 107.11	\$ 285.00	14.25
	+ Other	\$ 107.11	\$ 285.00	14.25
- Germany	Germany Total	\$ 86.84	\$ 2,242.00	28.50
	+ Other	\$ 86.84	\$ 2,242.00	28.50

المراجع:

- 1- Fundamentals of Database Systems: Ramez Elmasri & Shamkaul Navarthe; Addison-Wesely; 3rd edition. pp 845-855.
- 2- SQL Server 7.0 Data Warehousing Training Kit; Microsoft Press, November 1999.