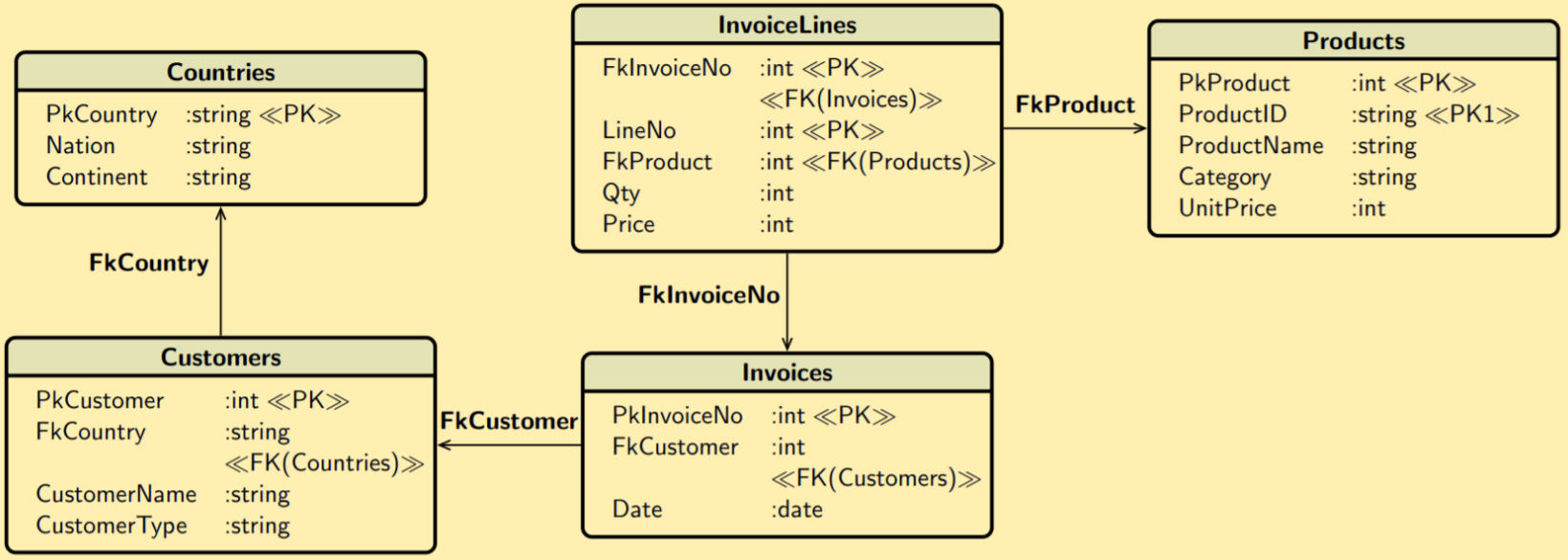
EXERCISE AT HOME

- Using JRS on the database TestStar, write SQL queries and check their Logical Query Plans for:



1. Number of distinct Customers by Product
2. Largest invoice revenue by Product
3. The percentage of revenue generated by the product over the total revenue of the customer by Customer and Product

SOLUTIONS

```
-- Number of distinct Customers by Product

SELECT FkProduct, COUNT(DISTINCT FkCustomer) AS NCustomer
FROM InvoiceLines, Invoices
WHERE FkInvoiceNo=PkInvoiceNo
GROUP BY FkProduct;



-- Largest invoice revenue by Product

WITH TotalByInvoice AS
   (SELECT FkInvoiceNo As InvoiceNo, SUM(Qty*Price) As TotalInvoice
   FROM InvoiceLines
   GROUP BY FkInvoiceNo)
SELECT FkProduct, MAX(TotalInvoice)
FROM InvoiceLines, TotalByInvoice
WHERE InvoiceNo=FkInvoiceNo
GROUP BY FkProduct;
```

SOLUTIONS

```
-- The percentage of revenue generated by the product over the total revenue
of the customer by Customer and Product

WITH a AS
    (SELECT FkCustomer, FkProduct, SUM(Price) AS Revenue
    FROM InvoiceLines, Invoices
    WHERE FkInvoiceNo = PkInvoiceNo
    GROUP BY FkCustomer, FkProduct),
b AS
    (SELECT FkCustomer, SUM(Price) AS Revenue
    FROM InvoiceLines, Invoices
    WHERE FkInvoiceNo = PkInvoiceNo
    GROUP BY FkCustomer)
SELECT a.FkCustomer, a.FkProduct, 100.0*a.Revenue/b.Revenue AS Ratio
FROM a, b
WHERE a.FkCustomer = b.FkCustomer
```

NEXT TOPICS

UP TO NOW

Data Warehouse: Data Models and DW Design and Implementation.

STARTING TODAY

Data Analysis Using SQL.

How to summarize data using SQL?

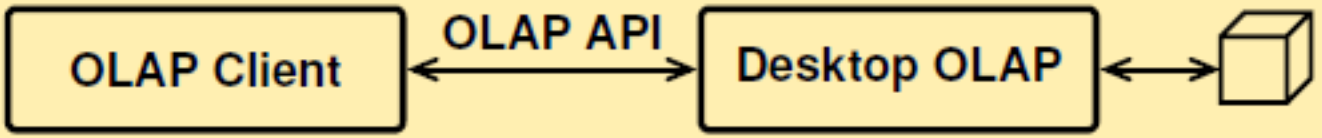
What if the query takes a long time to produce the answer?

OLAP SYSTEMS SOLUTIONS

OLAP refers to the technique of performing complex business multidimensional analysis over the data warehouse.

We will see **how report developers use SQL to write queries**!

OLAP SYSTEMS: SOLUTION 1

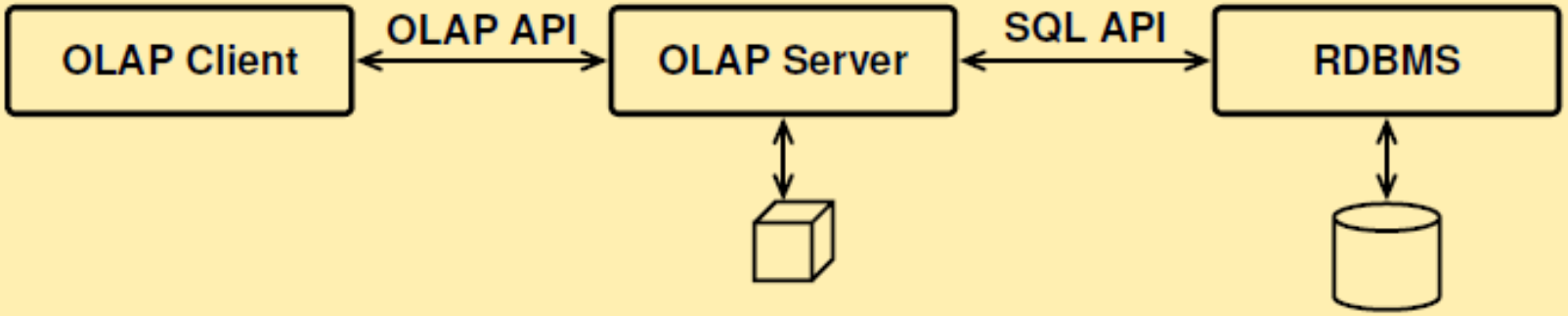


The **OLAP client** interacts with a local **DOLAP** system (**Desktop OLAP**) which manages small amount of data extracted from the **OLAP server**, the **Data server** or an **operational DBMS**. It a good choice for those who travel and move extensively, by using portable computers.

E.g., Excel Pivot Tables, Microsoft Power Pivot (Add-in of Excel)

We have seen Pivot Tables in Lesson 13.
You will see Power Pivot in the Lab of Data Science module

OLAP SYSTEMS: SOLUTION 2



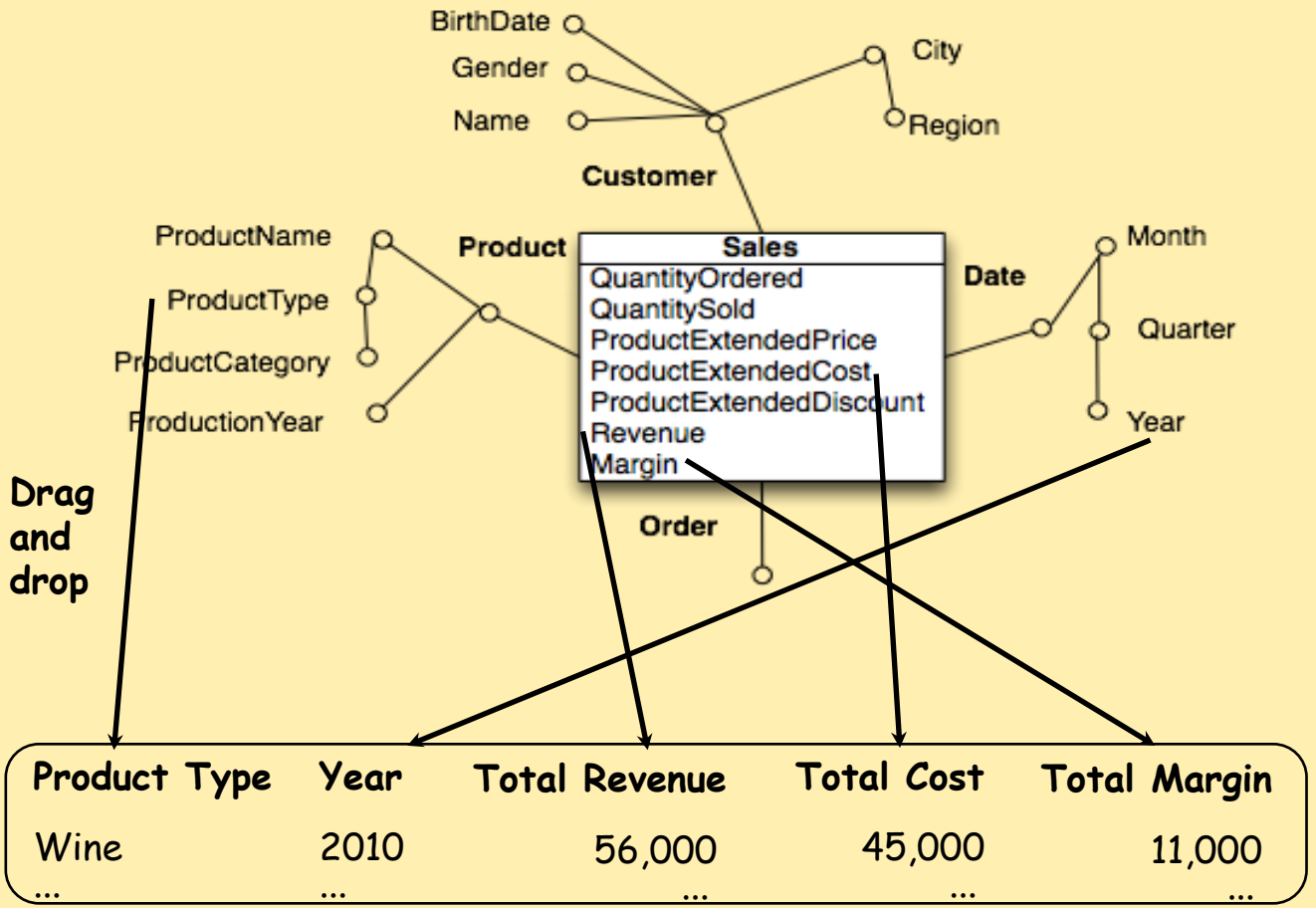
The **OLAP Client** interacts with an **OLAP Server**, that supports multidimensional data and operations, and can be one of the following type:

- **MOLAP**, which stores in the local memory the aggregates of the extended cube, using a specialized data structure. **A MOLAP server does not support SQL, but data cube query languages (MDX, DMX).**
- **ROLAP** which stores both the aggregates of the extended cube in the Data Server. ROLAP servers may also implement functionalities not supported in the SQL of the Data server.
- **HOLAP** which stores the data in the Data Server, and (part of) the aggregates of the extended cube in the local memory.

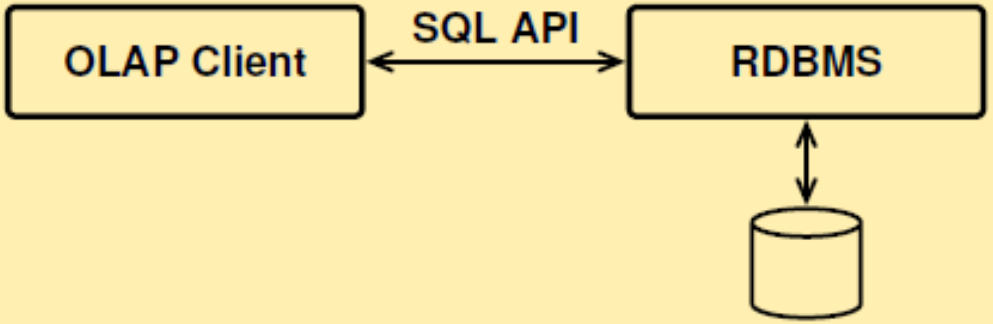
You will see these Solution 2 in the Lab of Data Science module

VISUAL REPORTING TOOLS

They allows a user or a developer to make data analysis and to build beautiful reports without any knowledge of SQL: Excel pivot tales, PowerPivot, Microstrategy, QlikView, etc



OLAP SYSTEMS: SOLUTION 3



The DW is managed by a specialized RDBMS (Relational **Data Server**)

The **OLAP Client** provides presentation and reporting tools to deal with data analysis and visualization, and interacts with the **Data Server** through SQL

We assume this solution in the rest of the course!

SIMPLE REPORTS WITH SQL

Sales(Customer, Product, Brand, Date, City, Region, Area, Quantity, Revenue, Margin)

Margin by Brand and by Product
Year 2009

Brand	Product	Revenue (€)	Margin (€)	Margin% (%)
B1	P1	2 100	273	13
	P2	3 720	624	17
	P3	15 300	1 803	12
B2	P4	12 600	756	6
	P5	22 500	2 196	10
	P6	48 300	4 496	9

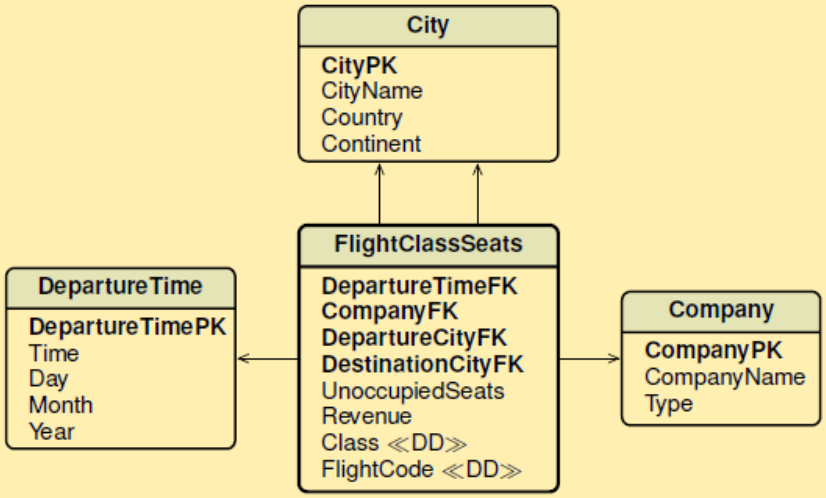
```
SELECT    Brand, Product, SUM(Revenue) AS Revenue,
          SUM(Margin) AS Margin,
          ROUND(100*SUM(Margin)/SUM(Revenue)) AS Margin%
FROM      Sales
WHERE     YEAR(Date) = 2009
GROUP BY  Brand, Product
ORDER BY  Brand, Product;
```

Slice

Rollup & drill-down

Pivoting

AIRLINE COMPANIES: DATA ANALYSIS

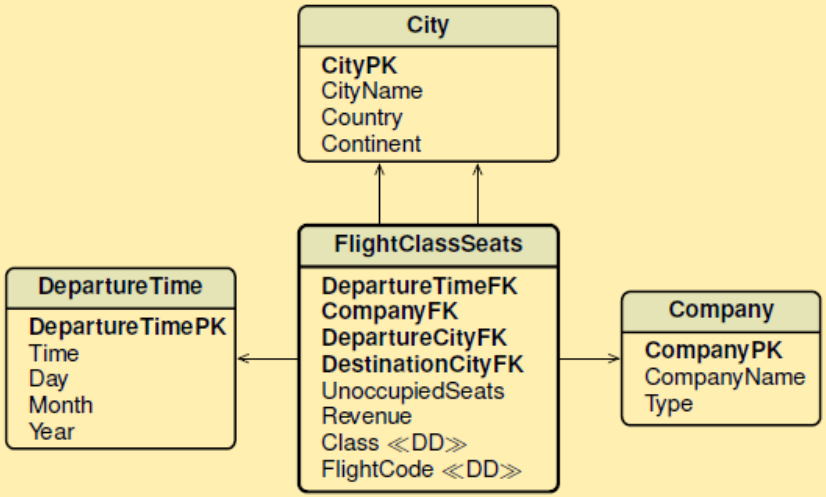


Requirements analysis
Number of unoccupied seats in a given year, by flight code, by company name (or type), by class, by departure time (time, day, month, year)
Number of unoccupied seats in a given class and year, by flight code, by company name, by class, by departure (destination) city (country, continent).
Number of unoccupied seats and revenue of the Alitalia company, by year, by month, by destination country.

Year = 2020

FlightCode	CompanyName	Class	Time	TotalUnoccupiedSeats
AZ2501	Alitalia	1st	8:00	250
BA471	British Airways	economy	10:00	302
...	...	...	...	...

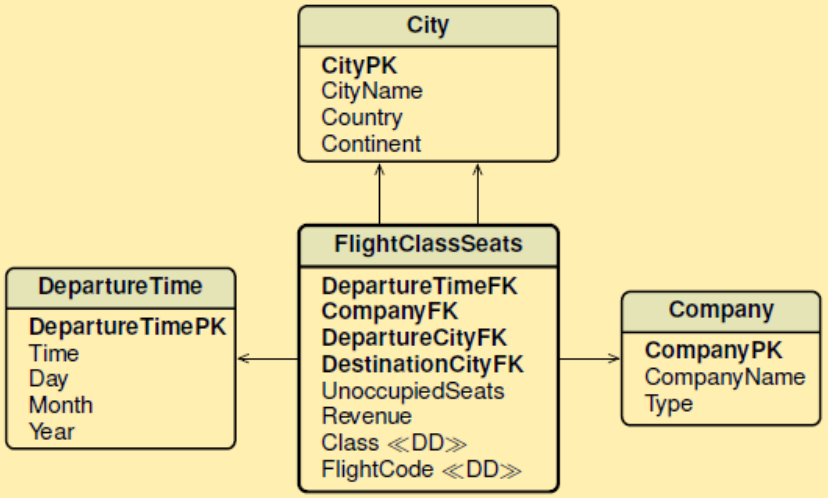
AIRLINE COMPANIES: DATA ANALYSIS



Requirements analysis
Number of unoccupied seats in a given year, by flight code, by company name (or type), by class, by departure time (time, day, month, year)
Number of unoccupied seats in a given class and year, by flight code, by company name, by class, by departure (destination) city (country, continent).
Number of unoccupied seats and revenue of the Alitalia company, by year, by month, by destination country.

```
SELECT   FlightCode, CompanyName, Class, Time, SUM(UnoccupiedSeats) As TotalUnoccupiedSeats
FROM     FlightClassSeats f, DepartureTime t, Company c
WHERE    f.DepartureTimeFK = t.DepartureTimePK AND f.CompanyFK = c.CompanyPK and year = 2020
GROUP BY FlightCode, CompanyName, Class, Time,
```

AIRLINE COMPANIES: DATA ANALYSIS



Requirements analysis
Number of unoccupied seats in a given year, by flight code, by company name (or type), by class, by departure time (time, day, month, year)
Number of unoccupied seats in a given class and year, by flight code, by company name, by class, by departure (destination) city (country, continent).
Number of unoccupied seats and revenue of the Alitalia company, by year, by month, by destination country.

```
SELECT  FlightCode, CompanyName, Class, City, SUM(UnoccupiedSeats) As TotalUnoccupiedSeats
FROM    FlightClassSeats f, DepartureTime t, City c
WHERE   f.DepartureTimeFK = t.DepartureTimePK AND f.DepartureCityFK = c.CityPK
          AND Class='Business' AND year = 2020
GROUP BY FlightCode, CompanyName, Class, City

SELECT  year, month, country, SUM(UnoccupiedSeats) As TotalUnoccupiedSeats,
                              SUM(Revenue) As TotalRevenue
FROM    FlightClassSeats f, DepartureTime t, City c
WHERE   f.DepartureTimeFK = t.DepartureTimePK AND f.DestinationCityFK= c.CityPK
            AND CompanyName='Alitalia'
GROUP BY year, month, country
```

ADDITIONAL CASES STUDIES

- Case Studies A.3-A.6
  - Conceptual design
  - Logical design
  - Data analysis (for some questions Analytics SQL is required)

- Solutions of A.1-A.6 are in the lecture notes:
  - RECOMMENDATION 1:
    - look at solutions only AFTER you have done the exercise!
  - RECOMMENDATION 2:
    - Compare the solution with your own and understand the differences!

SIMPLE REPORTS WITH SUBTOTALS

Sales(Customer, Product, Brand, Date, City, Region, Area, Quantity, Revenue, Margin)

Margin by Brand and by Product Year 2009				
Brand	**Product**	**Revenue (€)**	**Margin (€)**	**Margin% (%)**
B1	P1	2 100	273	13
	P2	3 720	624	17
	P3	15 300	1 803	12
B1	**Total**	**21 120**	**2 700**	**13**
B2	P4	12 600	756	6
	P5	22 500	2 196	10
	P6	48 300	4 496	9
B2	**Total**	**83 400**	**7 448**	**9**
Total		**104 520**	**10 148**	**10**

SIMPLE REPORTS WITH SUBTOTALS IN SQL

Sales(Customer, Product, Brand, Date, City, Region, Area, Quantity, Revenue, Margin)

Margin by Brand and by Product Year 2009				
Brand	**Product**	**Revenue (€)**	**Margin (€)**	**Margin% (%)**
B1	P1	2 100	273	13
	P2	3 720	624	17
	P3	15 300	1 803	12
B1	**Total**	**21 120**	**2 700**	**13**
B2	P4	12 600	756	6
	P5	22 500	2 196	10
	P6	48 300	4 496	9
B2	**Total**	**83 400**	**7 448**	**9**
Total		**104 520**	**10 148**	**10**

SIMPLE REPORTS WITH SUBTOTALS IN SQL

Sales(Customer, Product, Brand, Date, City, Region, Area, Quantity, Revenue, Margin)

```
SELECT    Brand, Product, SUM(Revenue) AS Revenue,
          SUM(Margin) AS Margin,
          ROUND(100*SUM(Margin)/SUM(Revenue)) AS Margin%
FROM      Sales
WHERE     YEAR(Date) = 2009
GROUP BY  Brand, Product

UNION ALL

SELECT    Brand, NULL AS Product, SUM(Revenue) AS Revenue,
          SUM(Margin) AS Margin,
          ROUND(100*SUM(Margin)/SUM(Revenue)) AS Margin%
FROM      Sales
WHERE     YEAR(Date) = 2009
GROUP BY  Brand

UNION ALL

SELECT    NULL AS Brand, NULL AS Product, SUM(Revenue) AS Revenue,
          SUM(Margin) AS Margin,
          ROUND(100*SUM(Margin)/SUM(Revenue)) AS Margin%
FROM      Sales
WHERE     YEAR(Date) = 2009
```

Margin by Brand and by Product
Year 2009

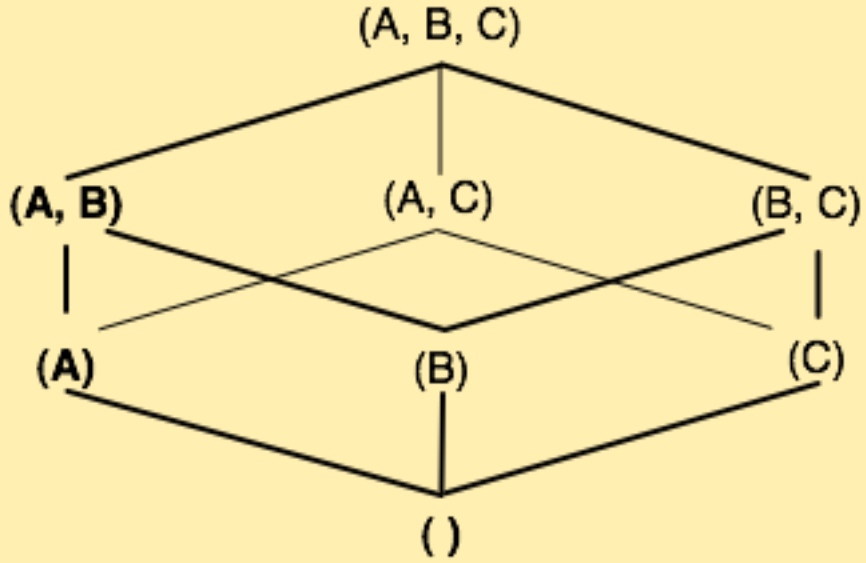
Brand	Product	Revenue (€)	Margin (€)	Margin% (%)
B1	P1	2 100	273	13
	P2	3 720	624	17
	P3	15 300	1 803	12
B1	**Total**	**21 120**	**2 700**	**13**
B2	P4	12 600	756	6
	P5	22 500	2 196	10
	P6	48 300	4 496	9
B2	**Total**	**83 400**	**7 448**	**9**
Total		**104 520**	**10 148**	**10**

GROUP BY ROLLUP(A,B)

Semantics: Union of 3 groupings:

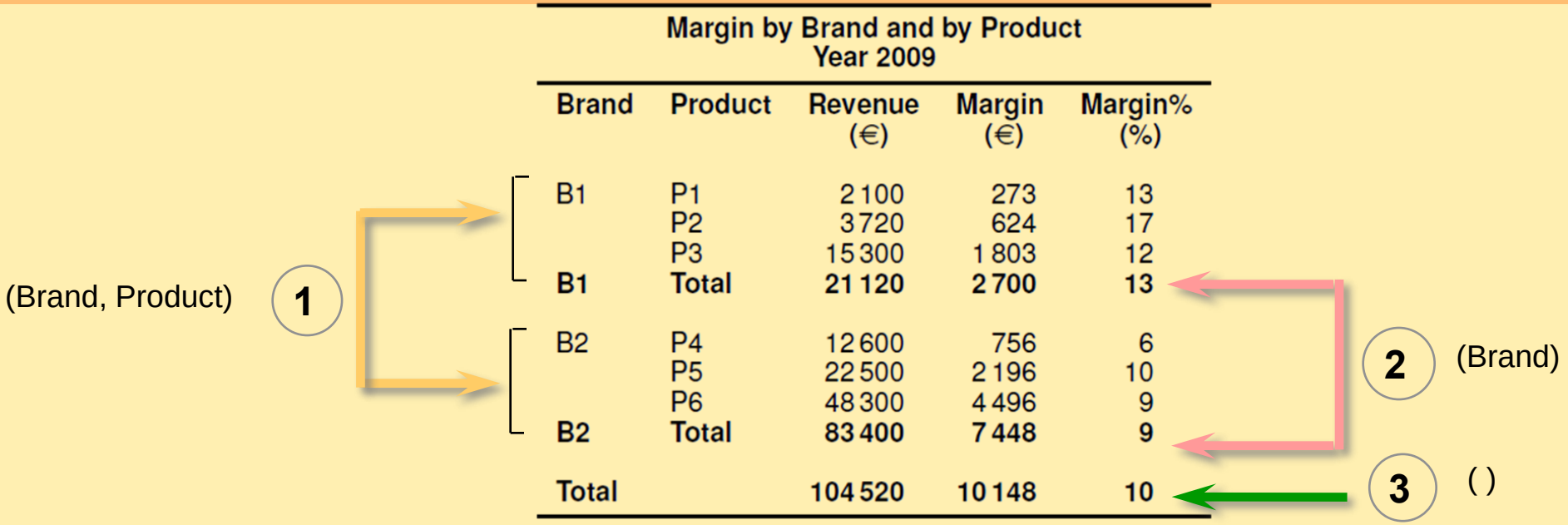
(A,B)
(A) subtotals
() totals

Attribute order is important!



ROLLUP compute one path through lattice

SIMPLE REPORTS WITH SUBTOTALS: ROLLUP




```
SELECT    Brand, Product, SUM(Revenue) AS Revenue,
          SUM(Margin) AS Margin,
          ROUND(100*SUM(Margin)/SUM(Revenue)) AS Margin%
FROM      Sales
WHERE     YEAR(Date) = 2009
GROUP BY  ROLLUP (Brand, Product)
ORDER BY  Brand, Product;
```

SIMPLE REPORTS WITH SUBTOTALS: CROSS-TABULATION

Product	Store			Total
	S1	S2	S3	
P1	300	500	50	**850**
P2	30	50	400	**480**
Total	**330**	**550**	**450**	**1330**

Margin by Brand and by Product
Year 2009

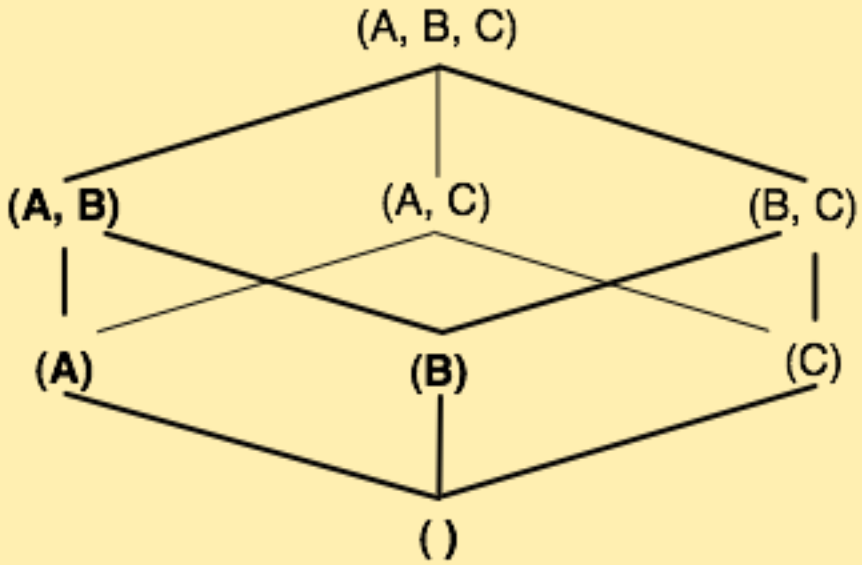
Brand	Product	Revenue (€)	Margin (€)	Margin% (%)
B1	P1	2 100	273	13
	P2	3 720	624	17
	P3	15 300	1 803	12
Total B1		**21 120**	**2 700**	**13**
B2	P4	12 600	756	6
	P5	22 500	2 196	10
	P6	48 300	4 496	9
Total B2		**83 400**	**7 448**	**9**
	Total P1	**2 100**	**273**	**13**
	Total P2	**3 720**	**624**	**17**
	Total P3	**15 300**	**1 803**	**12**
	Total P4	**12 600**	**756**	**6**
	Total P5	**22 500**	**2 196**	**10**
	Total P6	**48 300**	**4 496**	**9**
Total		**104 520**	**10 148**	**10**

GROUP BY CUBE(A,B)

Semantics: Union of 4 groupings:

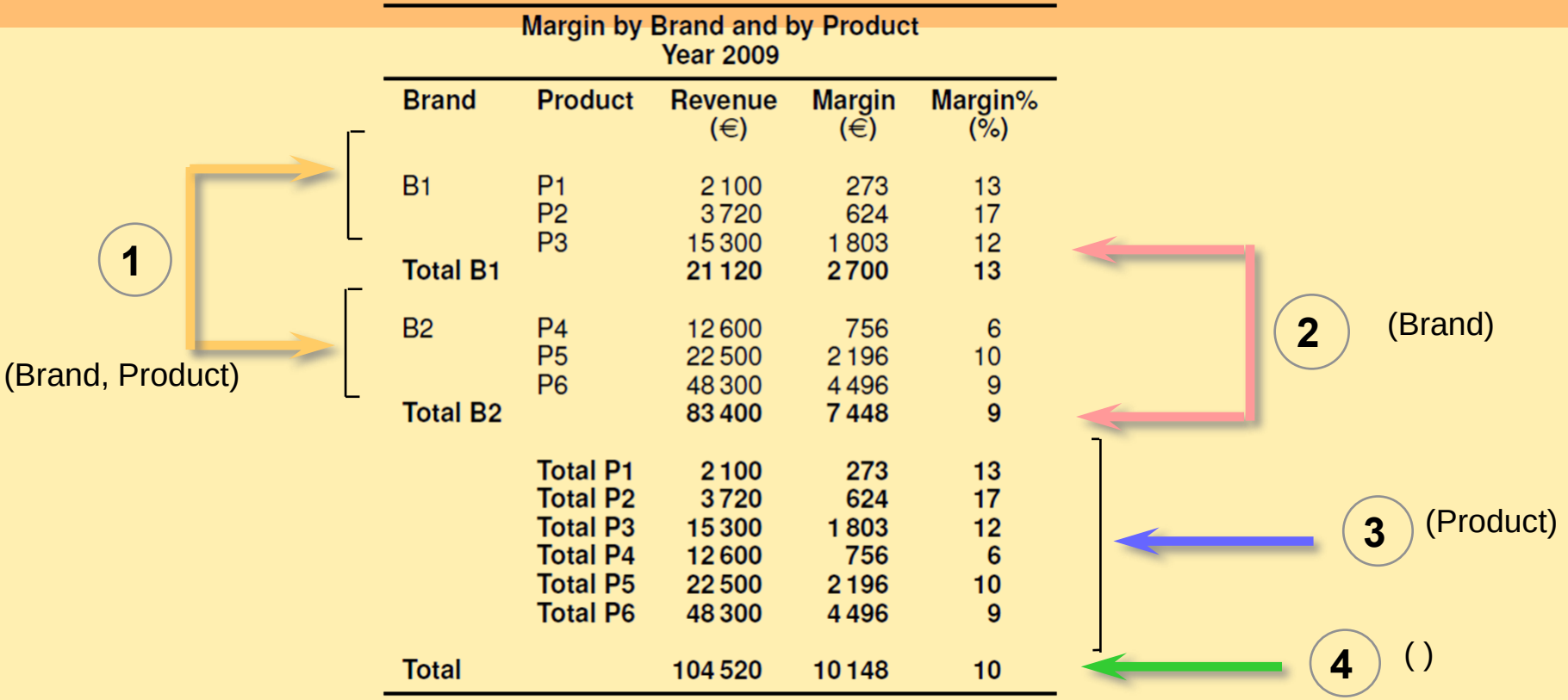
(A,B)
(A) subtotals
(B) subtotals
() totals

Attribute order doesn't matter



CUBE compute a sub-lattice

SIMPLE REPORTS WITH SUBTOTALS: CUBE



Margin by Brand and by Product
Year 2009

Brand	Product	Revenue (€)	Margin (€)	Margin% (%)
B1	P1	2 100	273	13
	P2	3 720	624	17
	P3	15 300	1 803	12
Total B1		**21 120**	**2 700**	**13**
B2	P4	12 600	756	6
	P5	22 500	2 196	10
	P6	48 300	4 496	9
Total B2		**83 400**	**7 448**	**9**
	Total P1	**2 100**	**273**	**13**
	Total P2	**3 720**	**624**	**17**
	Total P3	**15 300**	**1 803**	**12**
	Total P4	**12 600**	**756**	**6**
	Total P5	**22 500**	**2 196**	**10**
	Total P6	**48 300**	**4 496**	**9**
Total		**104 520**	**10 148**	**10**

```
SELECT    Brand, Product, SUM(Revenue) AS Revenue,
          SUM(Margin) AS Margin,
          ROUND(100*SUM(Margin)/SUM(Revenue)) AS Margin%
FROM      Sales
WHERE     YEAR(Date) = 2009
GROUP BY  CUBE (Brand, Product)
ORDER BY  Brand, Product;
```

PARTIAL ROLLUP AND CUBE

More than one ROLLUP and CUBE can be used in the GROUP BY

GROUP BY ROLLUP (*A*), **ROLLUP** (B, *C*)

Which groupings are computed? {(*A*), ()} X {(B, *C*), (B), ()}

= { (*A*, B, *C*), (*A*, B), (*A*), (B, *C*), (B), ( ) }

It is possible to compute only some groupings

GROUP BY *A*, **ROLLUP**(B, *C*)

computes the groupings: (*A*, B, *C*), (*A*, B), (*A*)

GROUP BY *A*, **CUBE**(B, *C*)

computes the groupings: (*A*, B, *C*), (*A*, B), (*A*, *C*), (*A*)

PARTIAL ROLLUP AND CUBE

It is possible to compute only some groupings

```
GROUP BY GROUPING SETS ( (A), (B, C) )
```

Foodmart datawarehouse

- RDBMS: Microsoft SQL Server
- **SQL Server: lds.di.unipi.it**
- **Login: dsd**

Must connect from .unipi.it (use VPN if your are outside)

- GUI:
  - SQL Server Management Studio
    - Win only
  - Azure Data Studio
    - Win, Linux, Mac OS