OLAP Operations

Helena Galhardas
DEI/IST

References

• A. Vaisman and E. Zimányi, Data Warehouse Systems: Design and Implementation, Springer, 2014 (chpt 3)
• J. Han and M. Kamber, Data Mining: Concepts and Techniques, Morgan Kaufmann, 2001 (chpt. 2)
• A. Wichert, H. Galhardas, SAD slides, MEIC/IST
Recap. the multidimensional model

- Multidimensional model enables one to view data from multiple perspectives and at several levels of detail

Example

**Dimensions**: perspectives used to analyze the data
- **Example**: A 3-dimensional cube for sales data with dimensions Product, Time, and Customer, and a measure Quantity

- **Attributes** describe dimensions
  - Product dimension may have attributes ProductNumber and UnitPrice (not shown)

- **Cells or facts** have associated numeric values called **measures**
  - Each cell of the data cube represents Quantity of units sold by category, quarter, and customer’s city
Hierarchies

- Allow viewing data at several granularities
  - Define a sequence of mappings relating lower-level, detailed concepts to higher-level ones
  - The lower level is called the child and the higher level is called the parent
  - The hierarchical structure of a dimension is called the dimension schema
  - A dimension instance comprises all members at all levels in a dimension

Example
- Hierarchies of the Product, Time, and Customer dimensions

Outline

- OLAP operations
OLAP Operations: definition

- Allows these perspectives and several levels of detail to be materialized by exploiting dimensions and their hierarchies
- Provide an interactive data analysis environment
- Supported by OLAP modules
  - Ex: Saiku (Pentaho)
OLAP Operations (2)

Q: Visualize the cube with the Time dimension in the x axis

Pivot

Q: Visualize data only for Paris or Lyon and quarters Q1 or Q2

Dice on city = 'Paris' or 'Lyon' and Quarter = 'Q1' or 'Q2'

Q: Visualize data only for Paris

Slice on city = 'Paris'

OLAP Operations (3)

Q: Compare the sales quantities in 2012 with those in 2011

Drill-across

Q: Compute the % change of sales between 2 years

Add measure
OLAP Operations (4)

Q: compute the total sales by quarter and city

\[ \text{sum}() \text{ by quarter and city} \]

Q: obtain the maximum sales by quarter and city

\[ \text{max}() \text{ by quarter and city} \]

- According to the authors of the book, agg. functions can be classified as:
  - cumulative: compute the measure value of a cell from several other cells (e.g., SUM; COUNT, AVG)
  - filtering: filter the members of a dimension that appears in the result (MIN, MAX); must compute not only the aggregated value but also determine the dimension members that belong to the result

OLAP Operations (5)

Q: see how monthly sales behave

3-month moving average()

Q: add data from Spain to the original cube

union
Algebra of OLAP Operations

- There is not yet a standard definition of OLAP operations in a similar way to the relational algebra.
- Many proposals of OLAP algebra in the literature.
- We adopt the one proposed in [Ciferri et al 2013]

Algebra of OLAP Operations - rollup

- Roll-up: aggregates measures along a dimension hierarchy (using an aggregate function) to obtain measures at a coarser granularity.
  
  \[ \text{ROLLUP(CubeName, (Dimension} \Rightarrow\text{ Level), AggFunction(Measure))} \]
Algebra of OLAP Operations – drill-down

- **Drill-down** moves from a more general level to a more detailed level in a hierarchy
  - \( \text{DRILLDOWN} (\text{CubeName}, (\text{Dimension} \rightarrow \text{Level})) \)

Algebra of OLAP Operations – sort

- **Sort** returns a cube where the members of a dimension have been sorted
  - \( \text{SORT} (\text{CubeName}, \text{Dimension}, \text{Expression} [\text{ASC} | \text{DESC}]) \)
  - where the members of Dimension are sorted according to the value of Expression
**Algebra of OLAP Operations – pivot**

- **Pivot (or rotate):** rotates the axes of a cube to provide an alternative presentation of data
  
  \[
  \text{PIVOT} \left( \text{CubeName}, (\text{Dimension} \rightarrow \text{Axis})^* \right) 
  \]
  
  where the axes are specified as \( \{X; Y; Z; X1; Y1; Z1; \ldots \} \).

**Algebra of OLAP Operations – slice**

- **Slice:** removes a dimension in a cube so a cube of \( n-1 \) dimensions is obtained from a cube of \( n \) dimensions
  
  \[
  \text{SLICE} \left( \text{CubeName}, \text{Dimension}, \text{Level} = \text{Value} \right) 
  \]
  
  Dimension will be dropped by fixing a single Value in the Level; other dimensions unchanged
  
  Slice supposes that the granularity of the cube is at the specified level of the dimension
Algebra of OLAP Operations – dice

- **Dice**: keeps the cells of a cube that satisfy a Boolean condition $\Phi$
  - $\text{DICE(CubeName, } \Phi\text{)}$
- $\Phi$ is a Boolean condition over dimension levels, attributes, and measures.

![Diagram of dice operation]

Q: Visualize data only for Paris or Lyon and quarters Q1 or Q2

$\text{DICE(Sales2012, } \{\text{Customer.City} = \text{"Paris" OR Customer.City = "Lyon" AND (Time.Quarter = "Q1" OR Time.Quarter = "Q2")}\} )$

Algebra of OLAP Operations – drill-across

- **Drill-across**: combines cells from two data cubes that have the same schema
  - $\text{DRILLACROSS(CubeName1, CubeName2, [Condition])}$

![Diagram of drill-across operation]

Q: Compare the sales quantities in 2012 with those in 2011

$\text{Sales2011-2012 } \bowtie\text{DRILLACROSS(Sales2011, Sales2012)}$
• **Add Measure**: adds new measures to a cube
  - `ADDMEASURE(CubeName, (NewMeasure = Expression)* )`
• **Drop measure**: Deletes a measure from a cube schema
  - `DROPMEASURE(CubeName, Measure*)`

Q: compute the % change of sales between 2 years

```
ADDMEASURE(Sales2011-2012, PercChange = (Quantity2011-Quantity2012)/Quantity2011)
```

Another ex:

- Computes the value of a cell by aggregating the measures of several nearby cells

```
ADDMEASURE(Sales2012, MovAvg = AVG(Quantity) OVER Time CELLS PRECEDING)
```
Aggregation functions in OLAP are also needed at the current granularity, that is without performing roll-up.

- **AggFunction(CubeName, Measure) [BY Dimension]**
- **Cumulative**: compute the measure value of a cell from several other cells; examples are SUM, COUNT, and AVG
- **Filtering**: Filters the members of a dimension that appear in the result; examples are MIN and MAX. Filtering functions compute not only the aggregated value, but also the members of the dimension that belong to the result

Another example: max sales by quarter and city
**Algebra of OLAP Operations – union, difference, drill-through**

- **Union** merges two cubes having the same schema but disjoint instances.
- Ex: If CubeSpain is a cube having the same schema as the original cube but containing only the sales to Spanish customers, we can perform:
- **Difference** removes the cells in a cube that belong to another one; the two cubes must have the same schema.
- **Drill-through** allows to move from data at the bottom level in a cube to data in the operational systems from which the cube was derived; Could be used when trying to determine the reason for outlier values in a data cube.

---

**Next Lecture**

- Conceptual Data Warehouse Design