Data Warehousing and OLAP Technology for Data Mining

— Chapter 3 —

Data Warehousing and OLAP Technology for Data Mining

- What is a data warehouse?
- A multi-dimensional data model
- Data warehouse architecture
- Data warehouse implementation

What is Data Warehouse?

- Defined in many different ways, but not rigorously.
 - A decision support database that is maintained separately from the organization's operational database
 - Support information processing by providing a solid platform of consolidated, historical data for analysis.
- "A data warehouse is a <u>subject-oriented</u>, <u>integrated</u>, <u>time-variant</u>, and <u>nonvolatile</u> collection of data in support of management's decision-making process."—W. H. Inmon
- Data warehousing:
 - The process of constructing and using data warehouses

Data Warehouse—Subject-Oriented

- Organized around major subjects, such as customer, product, sales.
- Focusing on the modeling and analysis of data for decision makers, not on daily operations or transaction processing.
- Provide a simple and concise view around particular subject issues by excluding data that are not useful in the decision support process.

Data Warehouse—Integrated

- Constructed by integrating multiple, heterogeneous data sources
 - relational databases, flat files, on-line transaction records
- Data cleaning and data integration techniques are applied.
 - Ensure consistency in naming conventions, encoding structures, attribute measures, etc. among different data sources
 - E.g., Hotel price: currency, tax, breakfast covered, etc.
 - When data is moved to the warehouse, it is converted.

Data Warehouse—Time Variant

- The time horizon for the data warehouse is significantly longer than that of operational systems.
 - Operational database: current value data.
 - Data warehouse data: provide information from a historical perspective (e.g., past 5-10 years)
- Every key structure in the data warehouse
 - Contains an element of time, explicitly or implicitly
 - But the key of operational data may or may not contain "time element".

Data Warehouse—Non-Volatile

- A physically separate store of data transformed from the operational environment.
- Operational update of data does not occur in the data warehouse environment.
 - Does not require transaction processing, recovery,
 and concurrency control mechanisms
 - Requires only two operations in data accessing:
 - initial loading of data and access of data.

Data Warehouse vs. Heterogeneous DBMS

- Traditional heterogeneous DB integration:
 - Query driven approach
 - When a query is posed to a client site, a meta-dictionary is used to translate the query into queries appropriate for individual heterogeneous sites involved, and the results are integrated into a global answer set
 - Complex information filtering, compete for resources
- Data warehouse: update-driven, high performance
 - Information from heterogeneous sources is integrated in advance and stored in warehouses for direct query and analysis

Data Warehouse vs. Operational DBMS

- OLTP (on-line transaction processing)
 - Major task of traditional relational DBMS
 - Day-to-day operations: purchasing, inventory, banking, manufacturing, payroll, registration, accounting, etc.
- OLAP (on-line analytical processing)
 - Major task of data warehouse system
 - Data analysis and decision making
- Distinct features (OLTP vs. OLAP):
 - User and system orientation: customer vs. market
 - Data contents: current, detailed vs. historical, consolidated
 - Database design: ER + application vs. star + subject
 - View: current, local vs. evolutionary, integrated
 - Access patterns: update vs. read-only but complex queries

OLTP vs. OLAP

	OLTP	OLAP
users	clerk, IT professional	knowledge worker
function	day to day operations	decision support
DB design	application-oriented	subject-oriented
data	current, up-to-date	historical,
	detailed, flat relational	summarized, multidimensional
	isolated	integrated, consolidated
usage	repetitive	ad-hoc
access	read/write	lots of scans
	index/hash on prim. key	
unit of work	short, simple transaction	complex query
# records accessed	tens	millions
#users	thousands	hundreds
DB size	100MB-GB	100GB-TB
metric	transaction throughput	query throughput, response

Why Separate Data Warehouse?

- High performance for both systems
 - DBMS— tuned for OLTP: access methods, indexing, concurrency control, recovery
 - Warehouse—tuned for OLAP: complex OLAP queries, multidimensional view, consolidation.
- Different functions and different data:
 - missing data: Decision support requires historical data which operational DBs do not typically maintain
 - data consolidation: DS requires consolidation (aggregation, summarization) of data from heterogeneous sources
 - data quality: different sources typically use inconsistent data representations, codes and formats which have to be reconciled

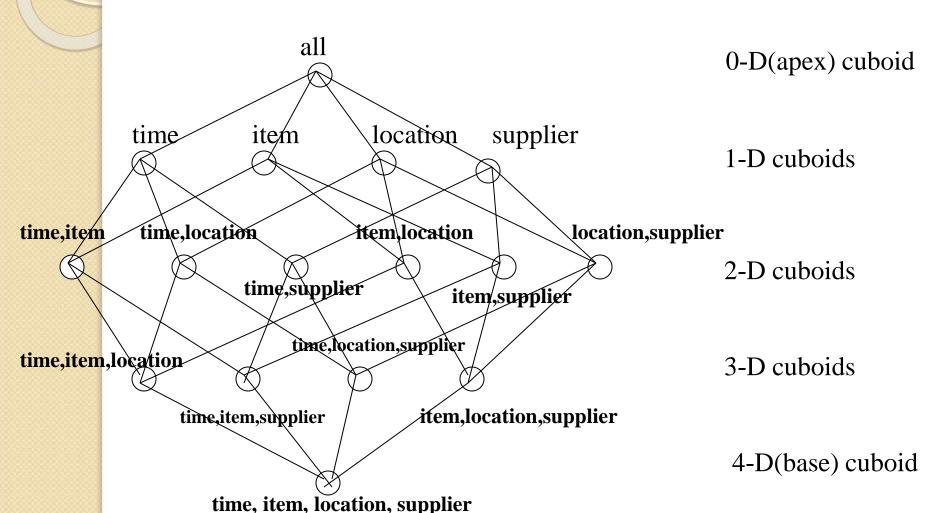
Data Warehousing and OLAP Technology for Data Mining

- What is a data warehouse?
- A multi-dimensional data model
- Data warehouse architecture
- Data warehouse implementation

From Tables and Spreadsheets to Data Cubes

- A data warehouse is based on a multidimensional data model which views data in the form of a data cube
- A data cube, such as sales, allows data to be modeled and viewed in multiple dimensions
 - Dimension tables, such as item (item_name, brand, type), or time(day, week, month, quarter, year)
 - Fact table contains measures (such as dollars_sold) and keys to each of the related dimension tables
- In data warehousing literature, an n-D base cube is called a base cuboid. The top most 0-D cuboid, which holds the highest-level of summarization, is called the apex cuboid. The lattice of cuboids forms a data cube.

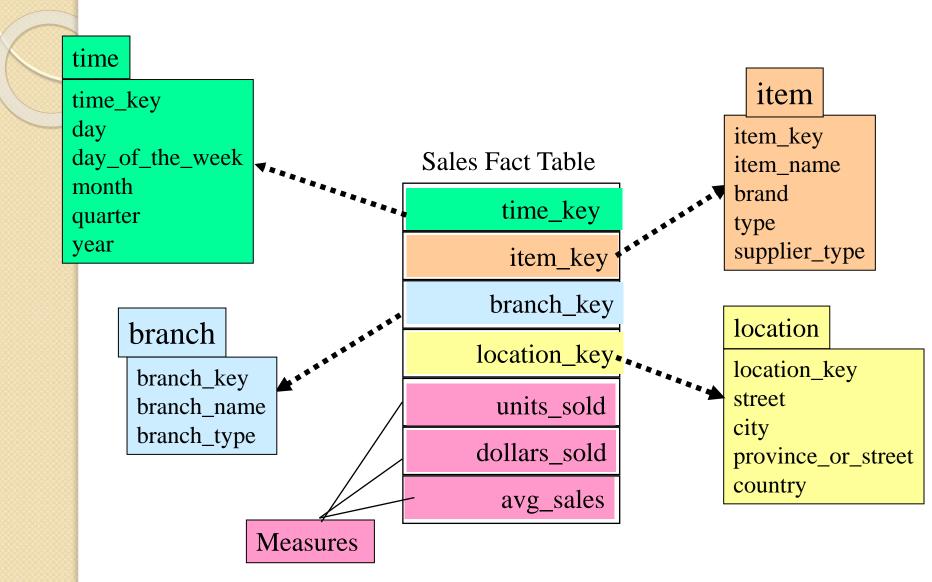
Cube: A Lattice of Cuboids



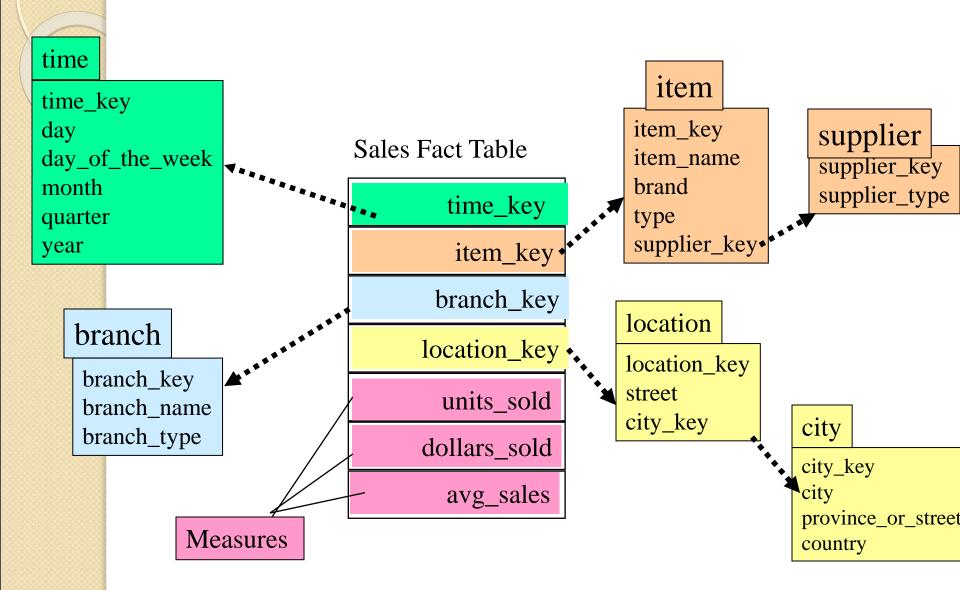
Conceptual Modeling of Data Warehouses

- Modeling data warehouses: dimensions & measures
 - Star schema: A fact table in the middle connected to a set of dimension tables
 - Snowflake schema: A refinement of star schema where some dimensional hierarchy is normalized into a set of smaller dimension tables, forming a shape similar to snowflake
 - <u>Fact constellations</u>: Multiple fact tables share dimension tables, viewed as a collection of stars, therefore called galaxy schema or fact constellation

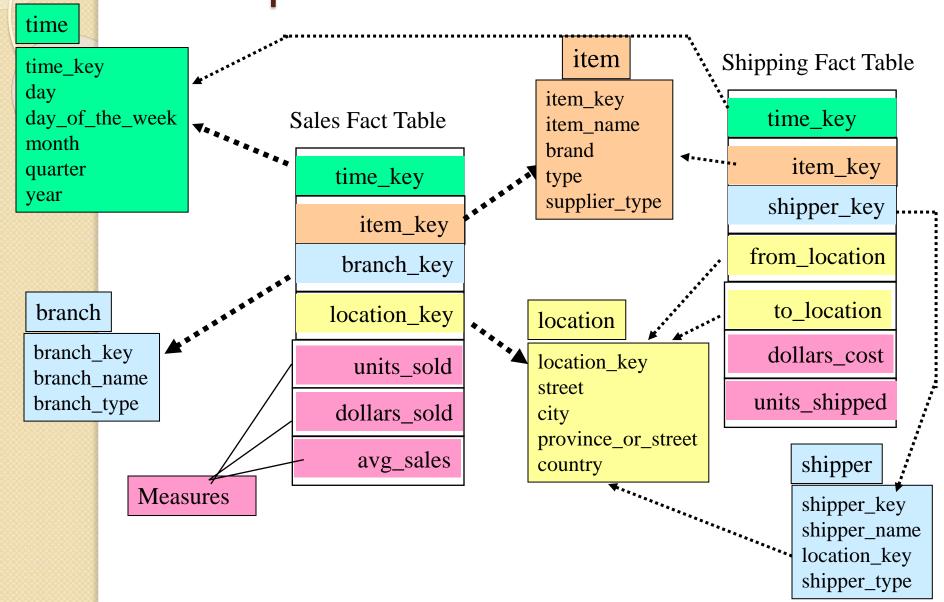
Example of Star Schema



Example of Snowflake Schema



Example of Fact Constellation



A Data Mining Query Language, DMQL: Language Primitives

- Cube Definition (Fact Table)
 define cube <cube_name> [<dimension_list>]:
 <measure list>
- Dimension Definition (Dimension Table)
 define dimension < dimension_name > as
 (<attribute_or_subdimension_list>)
- Special Case (Shared Dimension Tables)
 - First time as "cube definition"
 - define dimension <dimension_name> as
 <dimension_name_first_time> in cube
 <cube name first time>

Defining a Star Schema in DMQL

```
define cube sales_star [time, item, branch, location]:
    dollars_sold = sum(sales_in_dollars), avg_sales =
      avg(sales in dollars), units sold = count(*)
define dimension time as (time_key, day, day_of_week, month,
  quarter, year)
define dimension item as (item_key, item_name, brand, type,
  supplier type)
define dimension branch as (branch_key, branch_name,
  branch_type)
define dimension location as (location key, street, city,
  province_or_state, country)
```

Defining a Snowflake Schema in DMQL

```
define cube sales_snowflake [time, item, branch, location]:
    dollars sold = sum(sales in dollars), avg sales =
      avg(sales in_dollars), units_sold = count(*)
define dimension time as (time_key, day, day_of_week,
  month, quarter, year)
define dimension item as (item_key, item_name, brand,
  type, supplier (supplier key, supplier type))
define dimension branch as (branch_key, branch_name,
  branch type)
define dimension location as (location_key, street,
  city(city_key, province_or_state, country))
```

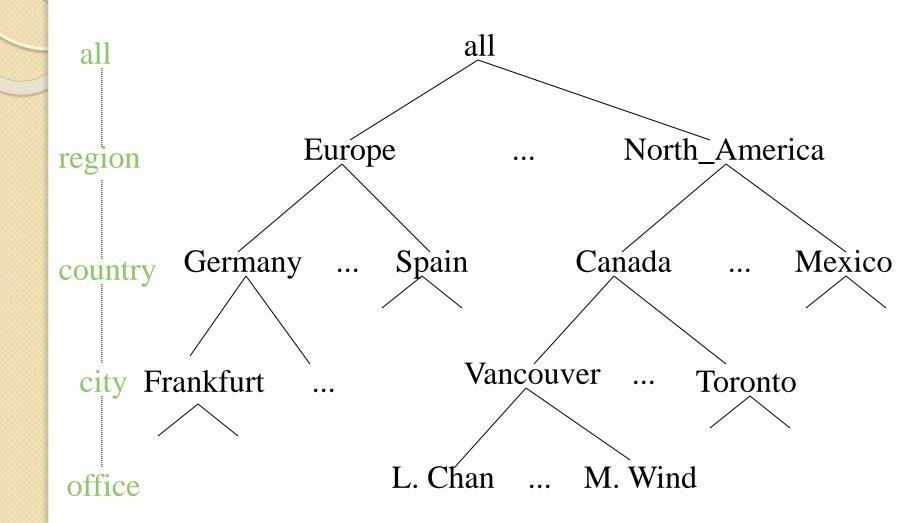
Defining a Fact Constellation in DMQL

```
define cube sales [time, item, branch, location]:
      dollars_sold = sum(sales_in_dollars), avg_sales = avg(sales_in_dollars),
        units sold = count(*)
define dimension time as (time_key, day_of_week, month, quarter, year)
define dimension item as (item_key, item_name, brand, type, supplier_type)
define dimension branch as (branch_key, branch_name, branch_type)
define dimension location as (location_key, street, city, province_or_state, country)
define cube shipping [time, item, shipper, from_location, to_location]:
      dollar cost = sum(cost in dollars), unit shipped = count(*)
define dimension time as time in cube sales
define dimension item as item in cube sales
define dimension shipper as (shipper_key, shipper_name, location as location in cube
   sales, shipper type)
define dimension from location as location in cube sales
define dimension to location as location in cube sales
```

Measures: Three Categories

- <u>distributive</u>: if the result derived by applying the function to *n* aggregate values is the same as that derived by applying the function on all the data without partitioning.
 - E.g., count(), sum(), min(), max().
- <u>algebraic</u>: if it can be computed by an algebraic function with *M* arguments (where *M* is a bounded integer), each of which is obtained by applying a distributive aggregate function.
 - E.g., avg(), min_N(), standard_deviation().
- <u>holistic</u>: if there is no constant bound on the storage size needed to describe a subaggregate.
 - E.g., median(), mode(), rank().

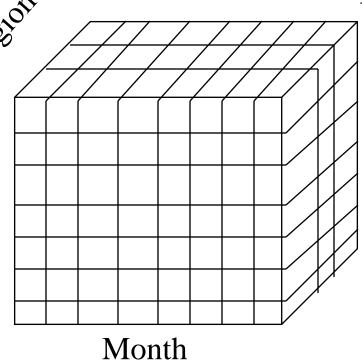
A Concept Hierarchy: Dimension (location)



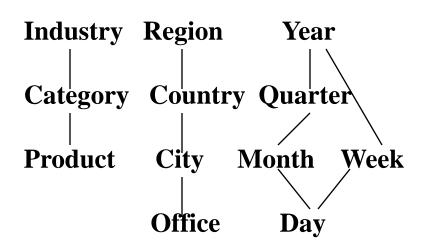
Multidimensional Data

Sales volume as a function of product, month, and region

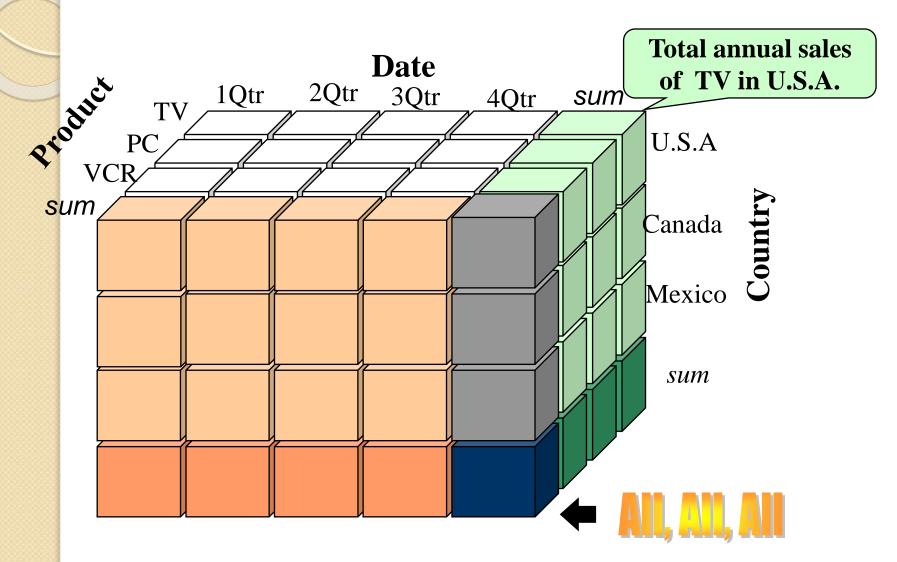
Dimensions: Product, Location, Time Hierarchical summarization paths



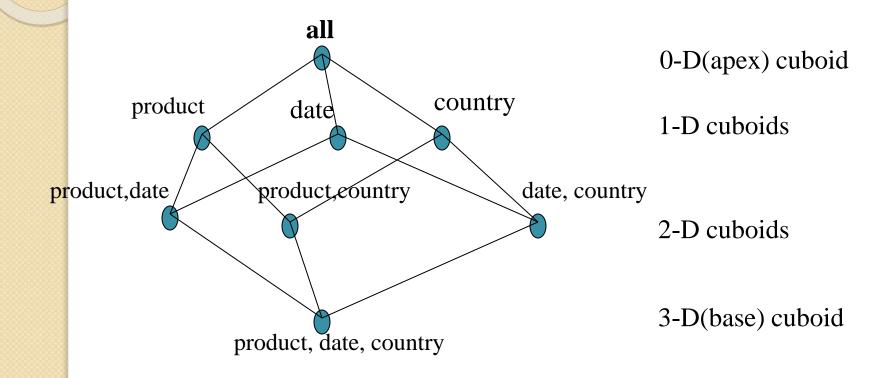
Product



A Sample Data Cube



Cuboids Corresponding to the Cube



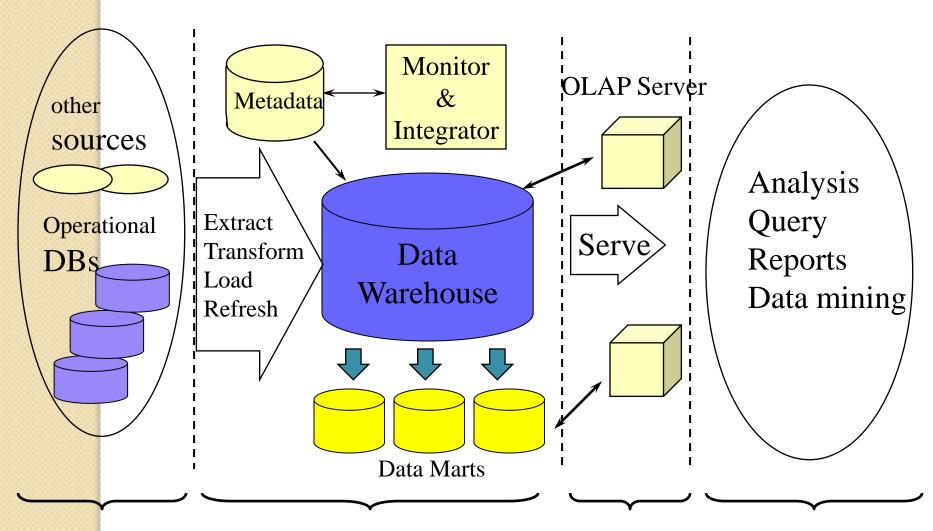
Typical OLAP Operations

- Roll up (drill-up): summarize data
 - by climbing up hierarchy or by dimension reduction
- Drill down (roll down): reverse of roll-up
 - from higher level summary to lower level summary or detailed data, or introducing new dimensions
- Slice and dice:
 - project and select
- Pivot (rotate):
 - reorient the cube, visualization, 3D to series of 2D planes.
- Other operations
 - drill across: involving (across) more than one fact table
 - drill through: through the bottom level of the cube to its back-end relational tables (using SQL)

Data Warehousing and OLAP Technology for Data Mining

- What is a data warehouse?
- A multi-dimensional data model
- Data warehouse architecture
- Data warehouse implementation

Multi-Tiered Architecture



Data Sources

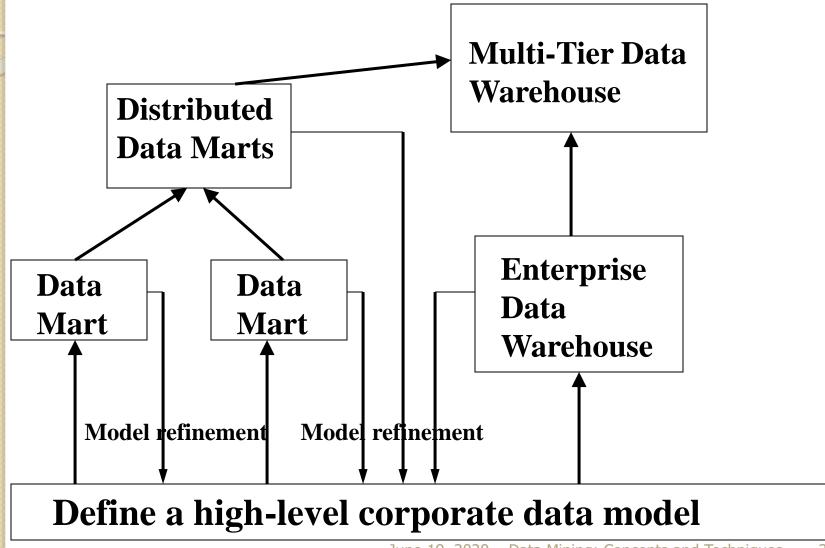
Data Storage

OLAP Engine Front-End Tools
June 19, 2020 Data Mining: Concepts and Techniques 30

Three Data Warehouse Models

- Enterprise warehouse
 - collects all of the information about subjects spanning the entire organization
- Data Mart
 - a subset of corporate-wide data that is of value to a specific groups of users. Its scope is confined to specific, selected groups, such as marketing data mart
 - Independent vs. dependent (directly from warehouse) data mart
- Virtual warehouse
 - A set of views over operational databases
 - Only some of the possible summary views may be materialized

Data Warehouse Development: A Recommended Approach



References (I)

- S. Agarwal, R. Agrawal, P. M. Deshpande, A. Gupta, J. F. Naughton, R. Ramakrishnan, and S. Sarawagi. On the computation of multidimensional aggregates. In Proc. 1996 Int. Conf. Very Large Data Bases, 506-521, Bombay, India, Sept. 1996.
- D. Agrawal, A. E. Abbadi, A. Singh, and T. Yurek. Efficient view maintenance in data warehouses. In Proc. 1997
 ACM-SIGMOD Int. Conf. Management of Data, 417-427, Tucson, Arizona, May 1997.
- R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan. Automatic subspace clustering of high dimensional data for data mining applications. In Proc. 1998 ACM-SIGMOD Int. Conf. Management of Data, 94-105, Seattle, Washington, June 1998.
- R. Agrawal, A. Gupta, and S. Sarawagi. Modeling multidimensional databases. In Proc. 1997 Int. Conf. Data Engineering, 232-243, Birmingham, England, April 1997.
- K. Beyer and R. Ramakrishnan. Bottom-Up Computation of Sparse and Iceberg CUBEs. In Proc. 1999 ACM-SIGMOD Int. Conf. Management of Data (SIGMOD'99), 359-370, Philadelphia, PA, June 1999.
- S. Chaudhuri and U. Dayal. An overview of data warehousing and OLAP technology. ACM SIGMOD Record, 26:65-74, 1997.
- OLAP council. MDAPI specification version 2.0. In http://www.olapcouncil.org/research/apily.htm, 1998.
- J. Gray, S. Chaudhuri, A. Bosworth, A. Layman, D. Reichart, M. Venkatrao, F. Pellow, and H. Pirahesh. Data cube: A relational aggregation operator generalizing group-by, cross-tab and sub-totals. Data Mining and Knowledge Discovery, 1:29-54, 1997.

References (II)

- V. Harinarayan, A. Rajaraman, and J. D. Ullman. Implementing data cubes efficiently. In Proc. 1996 ACM-SIGMOD Int. Conf. Management of Data, pages 205-216, Montreal, Canada, June 1996.
- Microsoft. OLEDB for OLAP programmer's reference version 1.0. In http://www.microsoft.com/data/oledb/olap, 1998.
- K. Ross and D. Srivastava. Fast computation of sparse datacubes. In Proc. 1997 Int. Conf. Very Large Data Bases, 116-125, Athens, Greece, Aug. 1997.
- K.A. Ross, D. Srivastava, and D. Chatziantoniou. Complex aggregation at multiple granularities. In Proc. Int.
 Conf. of Extending Database Technology (EDBT'98), 263-277, Valencia, Spain, March 1998.
- S. Sarawagi, R. Agrawal, and N. Megiddo. Discovery-driven exploration of OLAP data cubes. In Proc. Int. Conf. of Extending Database Technology (EDBT'98), pages 168-182, Valencia, Spain, March 1998.
- E. Thomsen. OLAP Solutions: Building Multidimensional Information Systems. John Wiley & Sons, 1997.
- Y. Zhao, P. M. Deshpande, and J. F. Naughton. An array-based algorithm for simultaneous multidimensional aggregates. In Proc. 1997 ACM-SIGMOD Int. Conf. Management of Data, 159-170, Tucson, Arizona, May 1997.

http://www.cs.sfu.ca/~han

