Data Warehousing

Lecture 3 Data Cube Technologies

CITS3401
CITS5504

Dr. Sirui Li

Computer Science and Software Engineering

School of Physics, Mathematics and Computing

Acknowledgement: The lecture slides are based on online sources.

Recap: Data Cube

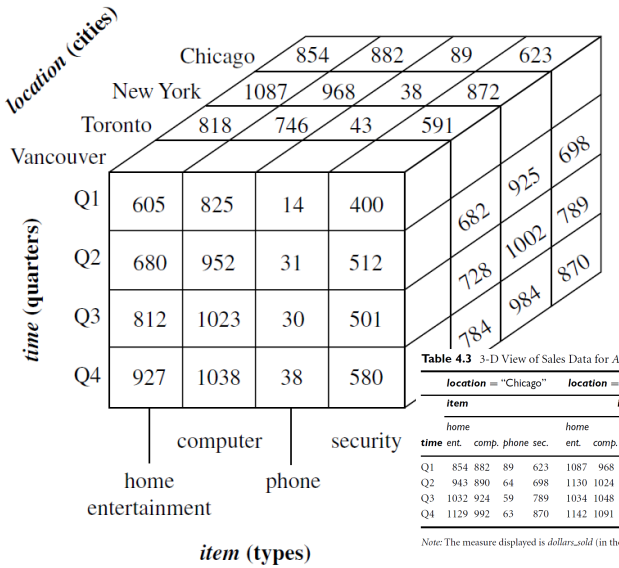
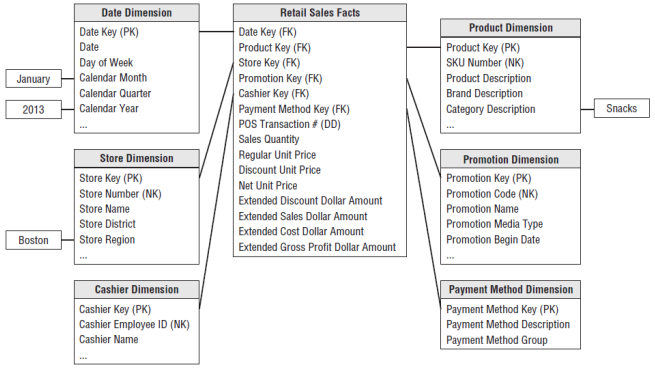


Table 4.3 3-D View of Sales Data for *AllElectronics* According to *time*, *item*, and *location*

	location = "Chicago"				*location* = "New York"				*location* = "Toronto"				*location* = "Vancouv
	item				*item*				*item*				*item*
	home				*home*				*home*				*home*
time	*ent.*	*comp.*	*phone*	*sec.*	*ent.*	*comp.*	*phone*	*sec.*	*ent.*	*comp.*	*phone*	*sec.*	*ent.*	*comp.*	*phone*	se
Q1	854	882	89	623	1087	968	38	872	818	746	43	591	605	825	14	40
Q2	943	890	64	698	1130	1024	41	925	894	769	52	682	680	952	31	51
Q3	1032	924	59	789	1034	1048	45	1002	940	795	58	728	812	1023	30	50
Q4	1129	992	63	870	1142	1091	54	984	978	864	59	784	927	1038	38	58

Note: The measure displayed is *dollars_sold* (in thousands).

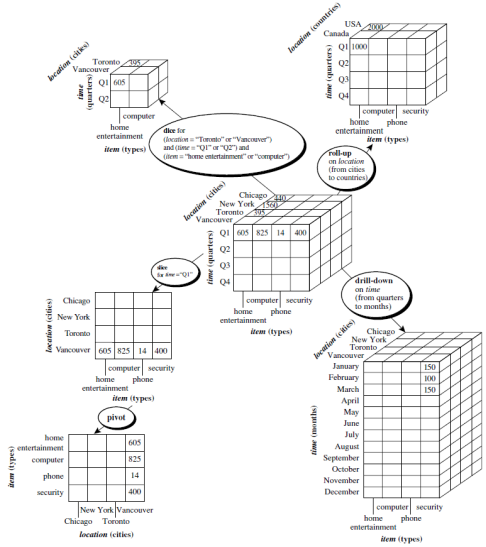
Recap: Fact Tables and Dimension Tables



Recap: Typical OLAP Operations

- **Roll up (drill up): summarise 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, visualisation, 3D to series of 2D planes.*
- **Other operations (aside)**
  - *drill across: involving (across) multiple fact tables*
  - *drill through: through the bottom level of the cube to its back-end relational tables (using SQL)*

Recap: Example of OLAP Operations



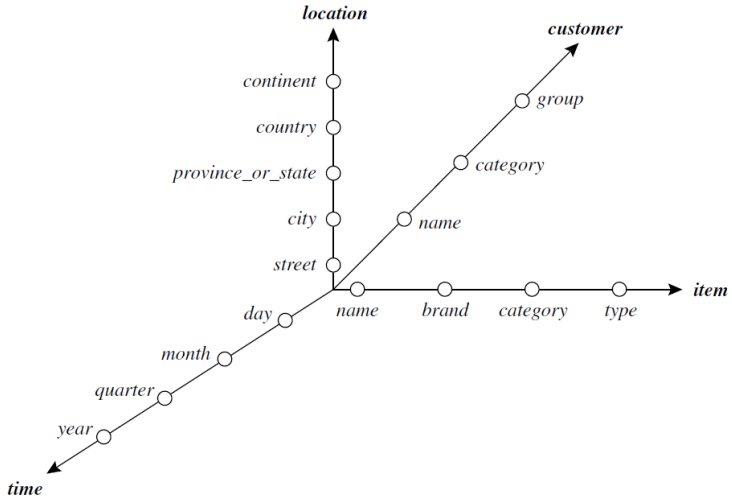
Multi-Dimensional Data Model

Concept of Hierarchies

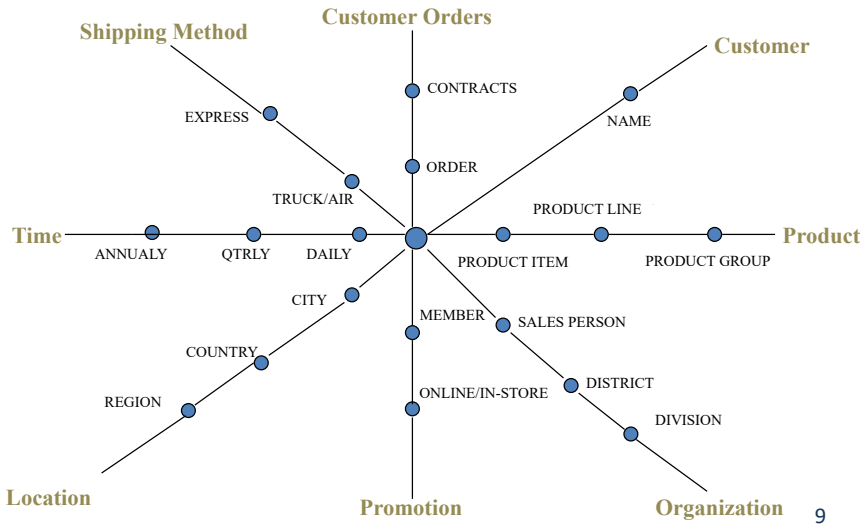
Starnet Query Model

- The querying of multidimensional databases can be based on a starnet model.
- A starnet model consists of radial lines emanating from a central point, where each line represents a concept hierarchy for a dimension.
- Each abstraction level in the hierarchy is called a footprint.
- These represent the granularities available for use by OLAP operations such as drill-down and roll-up.

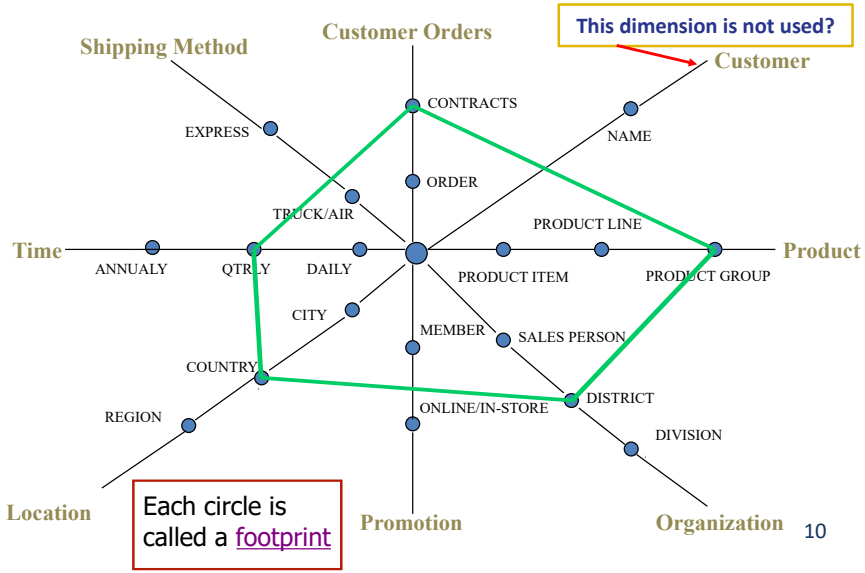
A Starnet Model of Business Queries



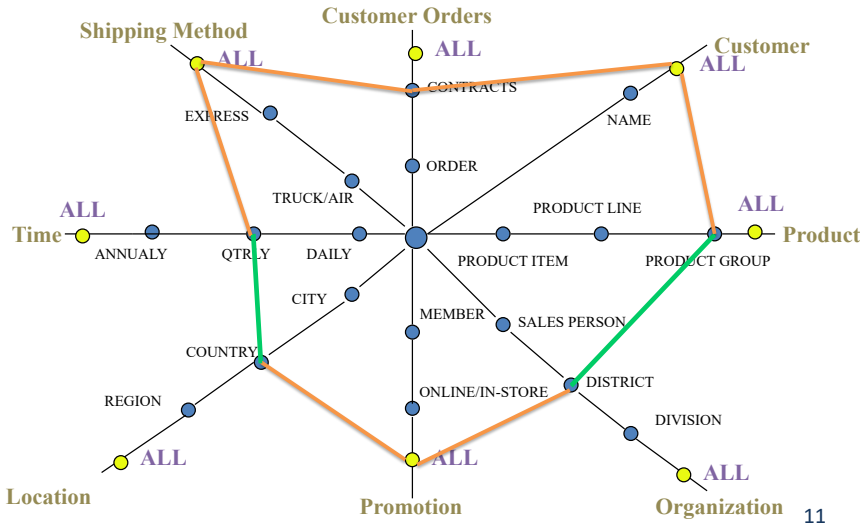
Granularity of viewing the data warehouse



Granularity of viewing the data warehouse



Granularity of viewing the data warehouse



Preferred mode of representation in assignment

Include ALL nodes

Multi-Dimensional Data Model

Data Warehouse Design Template

- Kimball's Four Steps

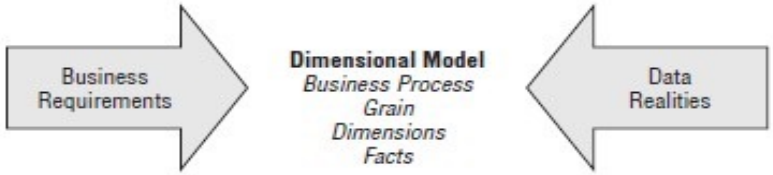
Data Warehouse Design Template

Kimball's four steps

- Identify a business process to model
  - E.g. orders, invoices, shipments, sales …
- Determine the grain of the business process
  - E.g. individual transactions, individual daily snapshots
- Choose the dimensions that apply to fact table rows
  - Example dimensions are time, item, customer, supplier, transaction type and status
- Identify the measure that populates fact table rows
  - Typical measures are numeric additive quantities like dollars_sold and units_sold

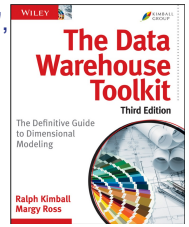
Kimball's four steps dimensional modeling

- Step 1: Identify the process being modelled.
- Step 2: Determine the grain at which facts will be stored.
- Step 3: Choose the dimensions.
- Step 4: Identify the numeric measures for the facts.



Running Example – Retail Sales

- **Grocery store chain recording POS retail sales**
  - Same example used in "The Data Warehouse Toolkit", Chapter 3
  - POS = Point of sale
    - Data collected by bar-code scanners at cash register
  - 100 grocery stores in 5 states
  - ~60,000 product SKUs
    - SKU = stock keeping unit
    - Represents an individual product
    - Some have UPCs (Universal Product Codes) assigned by manufacturer
    - Others don't (for example, produce, bakery, meat, floral)
  - Goal: understand impact of pricing & promotions on sales, profits
    - Promotions = coupons, discounts, advertisements, etc.



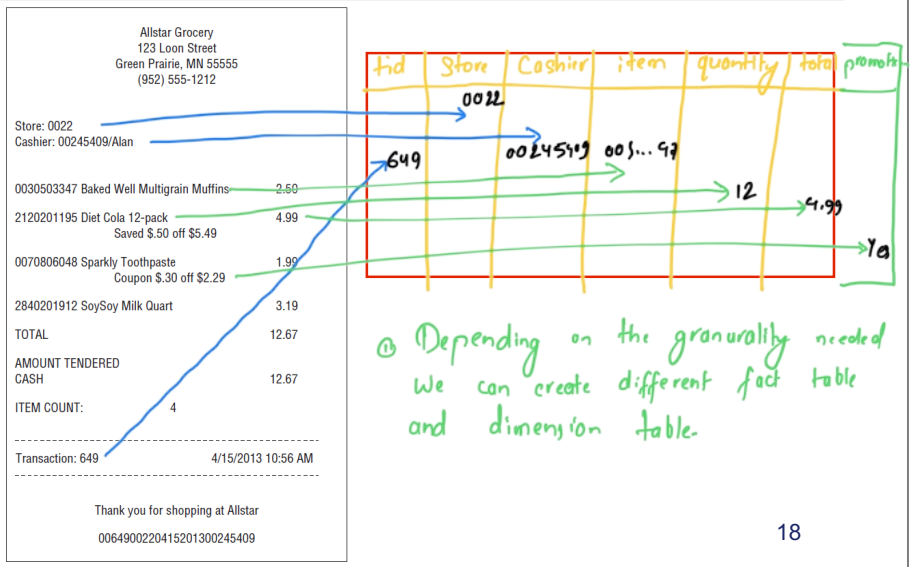
Step 1: Retail Sales Questions

- **What is the lift due to a promotion?**
  - Lift = gain in sales in a product because it's being promoted
  - Requires estimated baseline sales value
    - Could be calculated based on historical sales figures
- **Detect time shifting**
  - Customers stock up on the product that's on sale
  - Then they don't buy more of it for a long time
- **Detect cannibalisation**
  - Customers buy the promoted product instead of competing products
  - Promoting Brand A reduces sales of Brand B
- **Detect cross-sell of complementary products**
  - Promoting charcoal increases sales of lighter fluid
  - Promoting hamburger meat increases sales of hamburger buns
- **What is the profitability of a promotion?**
  - Considering promotional costs, discounts, lift, time shifting, cannibalisation, and cross-sell

Step 2: Grain of a fact table

- Grain of a fact table = the meaning of one fact table row
- Determines the maximum level of detail of the warehouse
- Example grain statements: (*one fact row represents a…*)
  - Line item from a cash register receipt
  - Boarding pass to get on a flight
  - Daily snapshot of inventory level for a product in a warehouse
  - Sensor reading per minute for a sensor
  - Student enrolled in a course
- Finer-grained fact tables:
  - are more expressive
  - have more rows
- Trade-off between performance and expressiveness
  - Rule of thumb: Errors in favor of expressiveness
  - Pre-computed aggregates can solve performance problems

A sample cash register receipt



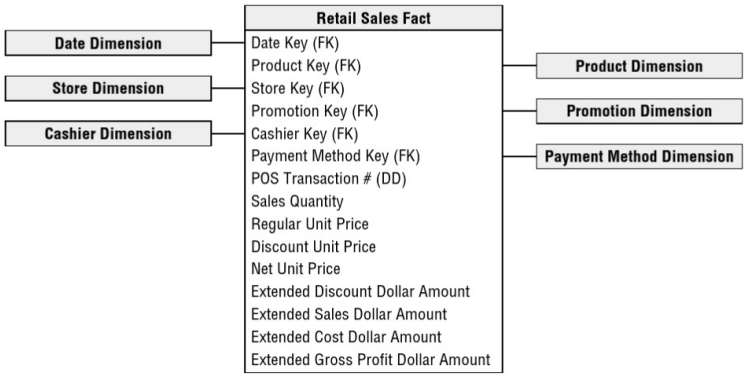
Step 3: Choosing dimensions

- Determine a candidate key based on the grain statement.
  - Example 1: a student enrolled in a course
    - (Course, Student, Term) is a candidate key
  - Example 2: line item from cash register receipt
    - (Transaction ID, Product SKU) is a candidate key
- Add other relevant dimensions that are functionally determined by the candidate key.
  - Example 1: Instructor and Classroom
    - Assuming each course has a single instructor!
  - Example 2: Store, Date, and Promotion

Step 4: Some numeric measures

- Quantity sold
- Total dollar sales
- Unit price
- Percentages (% discount)
- …

Measured facts



Sample Exam Question – Business Scenari

Important → Practice this

`Fitness Ninja` is multi-location health club chain that operates gyms Australia wide. They approached you to design and build a data warehouse. They have an existing Excel workbook that kept past five years of information about gym patrons' *every visit* to a gym in the `Fitness Ninja` health club chain:

- Name (First name, Last Name)
- Gender
- Driver's License
- Date of the visit
- Entry time
- Leaving time
- Visit as a member or not
- Location of the Gym visited, which is recorded in three separate columns
  - **Suburb**: e.g. Crawley
  - **City**: e.g. City of Claremont
  - **State**: e.g. WA
- Entry fee paid if not a member

They would like to design a data warehouse and also look into a graph database solution to meet their analytical needs. All questions in this examination refer to this particular business scenario.

Data Cube Technologies

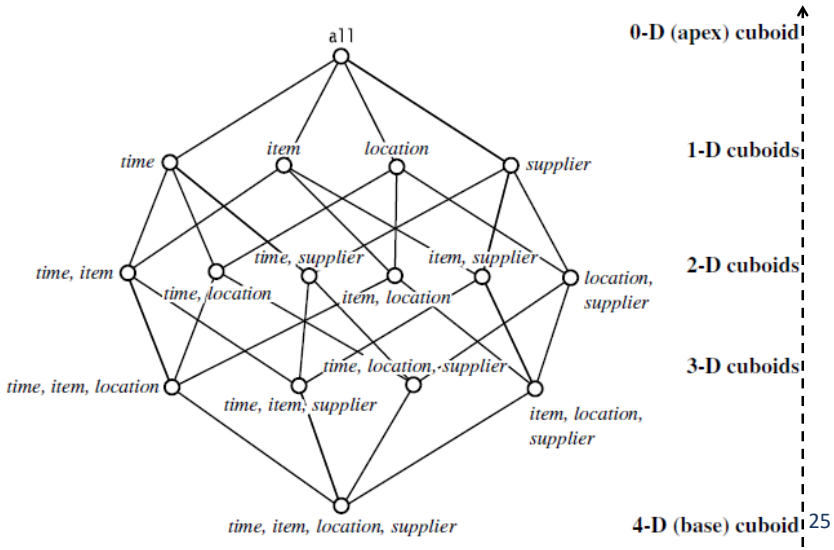
Data Cube

- Cuboids
- Types of Cells
- Types of Cubes

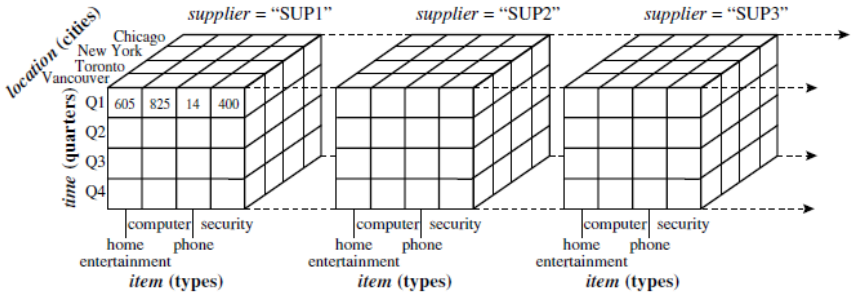
What is a Data Cube

- A data cube, is organised around a central theme, such as sales, allows data to be modelled and viewed in multiple dimensions
- Data cube is a metaphor for multi-dimentional data storage.
- The term hypercube is sometimes used, especially for data with more than three dimensions.
- A data cube is constructed from fact and dimension tables.
- A data cube is a lattice of cuboids.

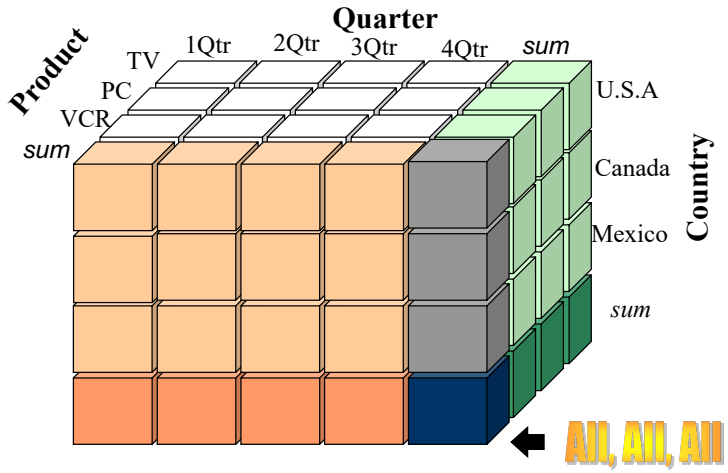
Cuboids in a Lattice



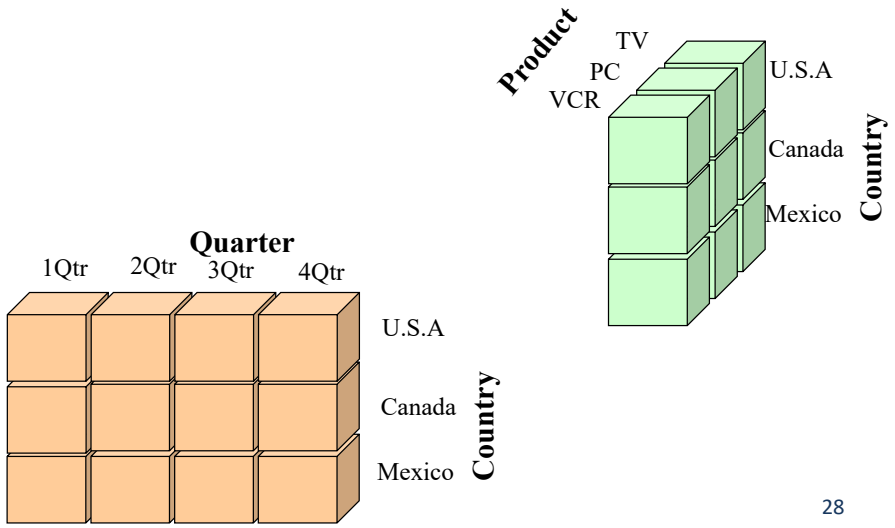
Base Cuboid



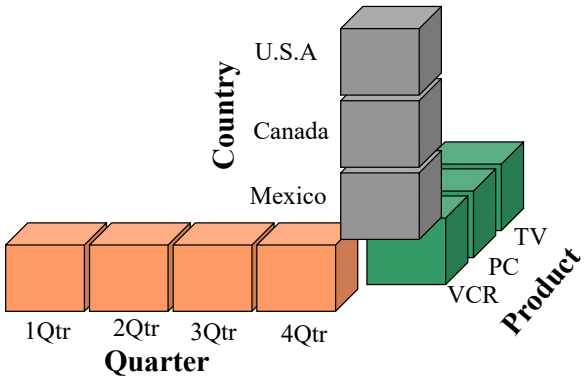
Sample Data Cube



Sample Data Cube: 2-D cubiods



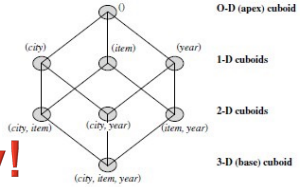
Sample Data Cube: 1-D cuboids



What is the total number of cuboids?

- A data cube is a lattice of cuboids. Suppose that you want to create a data cube for AllElectronics sales that contains the following: *city*, *item*, *year*, and *sales* in dollars.
- Possible queries such as the following:
  - "Compute the sum of sales, grouping by city and item."
  - "Compute the sum of sales, grouping by city."
  - "Compute the sum of sales, grouping by item."

Curse of dimensionality!



Multi-Dimensional Data Model

Answering Queries with Data Cubes in SQL

Standard Operations to Answer Queries

Measurements
- Which fact(s) should be reported?

Filters
- What slice(s) of the cube should be used?

Grouping attributes
- How finely should the cube be diced?
- Each dimension is either:
  (a) A grouping/categorical/discrete attribute
  (b) Aggregated over ("Rolled up" into a single total)

n dimensions → 2^n sets of grouping attributes

Aggregation = projection to a lower-dimensional subspace

Efficient Processing of OLAP Queries

- Given four materialised cuboids, the query to be processed is on {brand, province or state}, with the selection constant "year = 2010."
  - cuboid 1: {year, item name, city}
  - cuboid 2: {year, brand, country}
  - cuboid 3: {year, brand, province or state}
  - cuboid 4: {item name, province or state}, where year = 2010
- Which one to choose?
  - Cuboids 1, 3, and 4 can be used to process the query because
    - they have the same set or a superset of the dimensions in the query,
    - the selection clause in the query can imply the selection in the cuboid, and
    - the abstraction levels for the item and location dimensions in these cuboids are at a finer level than brand and province or state, respectively.

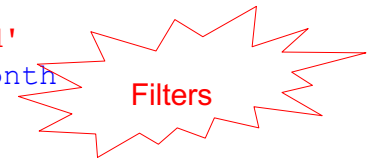
Using Cube in Queries

- Queries with Data Cube in SQL Server

```
SELECT month, state, SUM (amount)
FROM SALES
CUBE BY month, state
```


Creating Cross Tab with SQL


```
SELECT state, month, SUM(quantity)
FROM sales
WHERE color = 'Red'
GROUP BY state, month
```



	VIC	NSW	WA	Total
Jul	45	33	30	108
Aug	50	36	42	128
Sep	38	31	40	109
Total	133	100	112	345

Cross Tab Report

What about the totals

- SQL aggregation query with GROUP BY does not produce subtotals, totals
- Our cross-tab report is incomplete.

Autos Sold

	VIC	NSW	WA	Total
Jul	45	33	30	?
Aug	50	36	42	?
Sep	38	31	40	?
Total	?	?	?	?

State	Month	SUM
VIC	Jul	45
VIC	Aug	50
VIC	Sep	38
NSW	Jul	33
NSW	Aug	36
NSW	Sep	31
WA	Jul	30
WA	Aug	42
WA	Sep	40

One solution: a big UNION ALL

	VIC	NSW	WA	Total
Jul	45	33	30	?
Aug	50	36	42	?
Sep	38	31	40	?
Total	?	?	?	?

Original Query

```
SELECT state, month, SUM(quantity)
FROM sales
WHERE color = 'Red'
GROUP BY state, month
UNION ALL
```

State Subtotals

```
SELECT state, 'ALL', SUM(quantity)
FROM sales
WHERE color = 'Red'
GROUP BY state
UNION ALL
```

Month Subtotals

```
SELECT 'ALL', month, SUM(quantity)
FROM sales
WHERE color = 'Red'
GROUP BY month
UNION ALL
```

Overall Total

```
SELECT 'ALL', 'ALL', SUM(quantity)
FROM sales
WHERE color = 'Red'
```

"UNION ALL" on > 2 attributes ??

One solution: a big UNION ALL

	State	Month	Sales
1	VIC	Jul	45
2	VIC	Aug	50
3	VIC	Sep	38
4	NSW	Jul	33
5	NSW	Aug	36
6	NSW	Sep	31
7	WA	Jul	30
8	WA	Aug	42
9	WA	Sep	40

```
SELECT [state], [month], SUM(sales) as total
  FROM [Tut_Wei_db1].[dbo].[Lecture3Demo]
  GROUP BY [state], [month]
UNION ALL
SELECT [state], 'ALL', SUM(sales) as total
  FROM [Tut_Wei_db1].[dbo].[Lecture3Demo]
  GROUP BY [state]
UNION ALL
SELECT 'ALL', [month], SUM(sales) as total
  FROM [Tut_Wei_db1].[dbo].[Lecture3Demo]
  GROUP BY [month]
UNION ALL
SELECT 'ALL', 'ALL', SUM(sales) as total
  FROM [Tut_Wei_db1].[dbo].[Lecture3Demo]
```

	state	month	total
1	NSW	Aug	36
2	VIC	Aug	50
3	WA	Aug	42
4	NSW	Jul	33
5	VIC	Jul	45
6	WA	Jul	30
7	NSW	Sep	31
8	VIC	Sep	38
9	WA	Sep	40
10	NSW	ALL	100
11	VIC	ALL	133
12	WA	ALL	112
13	ALL	Aug	128
14	ALL	Jul	108
15	ALL	Sep	109
16	ALL	ALL	345

A better solution

- "UNION ALL" solution gets cumbersome with more than 2 grouping attributes
- n grouping attributes → 2^n parts in the union
- OLAP extensions added to SQL 99 are more convenient
  - CUBE,ROLLUP

```
SELECT state, month, SUM(quantity)
FROM sales
GROUP BY CUBE(month, state)
WHERE color = 'Red'
```


Results of the Cube Query

	VIC	NSW	WA	Total
Jul	45	33	30	108
Aug	50	36	42	128
Sep	38	31	40	109
Total	133	100	112	345

Notice the use of NULL for totals

State	Month	SUM(quantity)
VIC	Jul	45
VIC	Aug	50
VIC	Sep	38
VIC	NULL	133
NSW	Jul	33
NSW	Aug	36
NSW	Sep	31
NSW	NULL	100
WA	Jul	30
WA	Aug	42
WA	Sep	40
WA	NULL	112
NULL	Jul	108
NULL	Aug	128
NULL	Sep	109
NULL	NULL	345

Subtotals for months

total

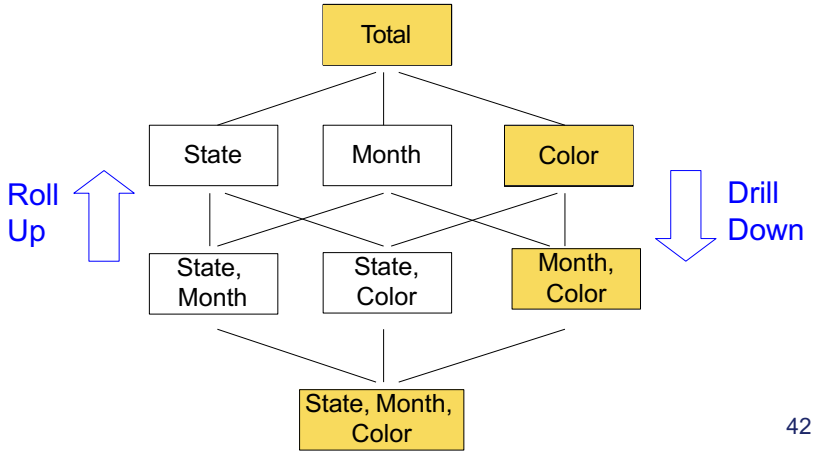
ROLLUP vs. CUBE

- CUBE computes entire lattice
- ROLLUP computes one path through lattice
  - Order of GROUP BY list matters
  - Groups by all prefixes of the GROUP BY list

- GROUP BY ROLLUP(A,B,C)
  - A,B,C
  - (A,B) subtotals
  - (A) subtotals
  - Total
- GROUP BY CUBE(A,B,C)
  - A,B,C
  - Subtotals for the following: (A,B), (A,C), (B,C), (A), (B), (C)
  - Total

Data Cube Lattice

```
SELECT color, month, state, SUM(quantity)
FROM sales
GROUP BY ROLLUP(color,month,state)
```



References

- Readings
  - Han et al. Chapter 5
  - Kimball et al. Chapter 3
  - What is cross-tabulation?
  - How to implement one-to-one, one-to-many and many-to-many relationships of an ER model?

Copyright Notice

Copyright Notice