Data cube: a relational aggregation operator generalizing group-by, cross-tab, and sub-totals

Advanced Topics in Databases

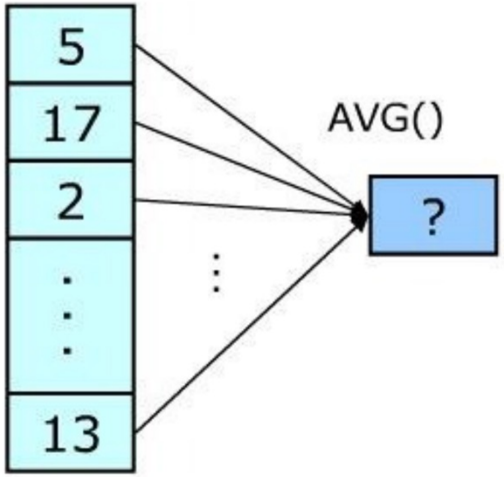
Outline

- Reminding:
  - Relational Aggregate Operators
  - Relational GROUP BY Operator
- Problems with GROUP BY
  - Histograms
  - Roll-Up Reports
  - Cross-Tabs
- CUBE Operator
  - Think of the N-dimensional Cube
  - An Example
  - Syntax
  - Functional Aggregation
  - Computing the Cube

Relational Aggregate Operators

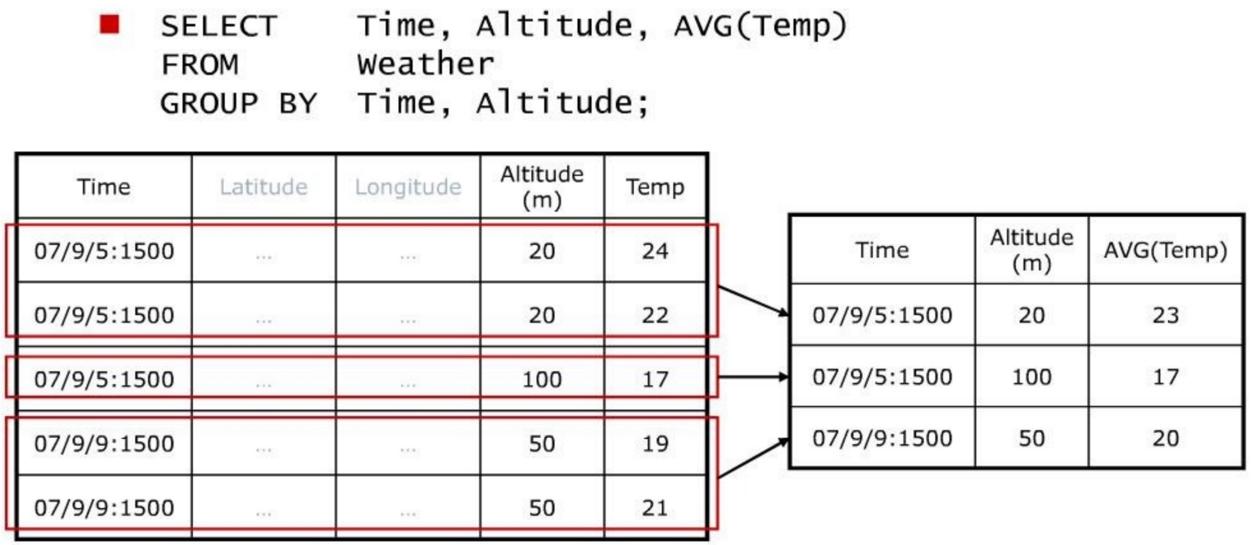
- SQL has several aggregate operators:
  - SUM(), MIN(), MAX(), COUNT(), AVG()
  - Some systems extend this with many other:
    - Stat functions, financial functions...
    - i.e. RANK(), N_TILE(), RATIO_TO_TOTAL()
- The basic idea is:
  - Combine all values in a column into a scalar value
- Syntax

  -
    ```
     SELECT AVG(Temp)
    FROM Weather;
    ```



Relational GROUP BY Operator

- Aggregation functions return a single value.
- Using the GROUP BY operator SQL can create a table with several tuples indexed by a set of attributes.
- Example:



```
SELECT     Time, Altitude, AVG(Temp)
FROM       Weather
GROUP BY   Time, Altitude;
```

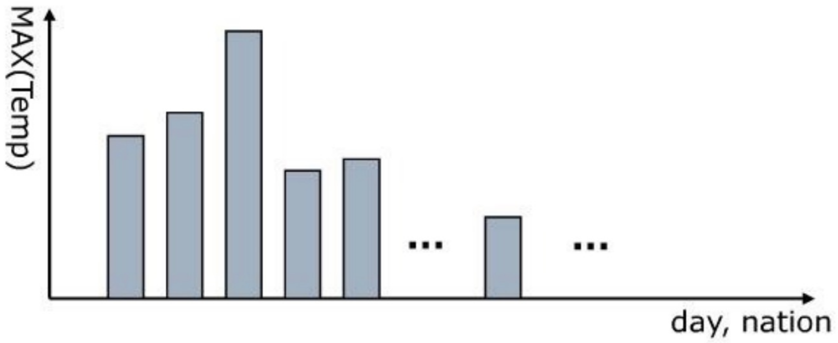
Time	Latitude	Longitude	Altitude (m)	Temp
07/9/5:1500	...	...	20	24
07/9/5:1500	...	...	20	22
07/9/5:1500	...	...	100	17
07/9/9:1500	...	...	50	19
07/9/9:1500	...	...	50	21

Time	Altitude (m)	AVG(Temp)
07/9/5:1500	20	23
07/9/5:1500	100	17
07/9/9:1500	50	20

Problems with GROUP BY - Histograms

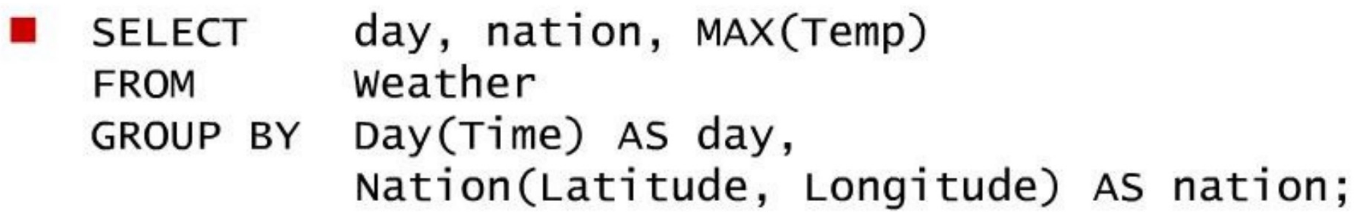
- Users want histograms
    - Suppose:
        - Day(): time –> day
        - Nation(): latitude & longitude -> name of country

```
SELECT     day, nation, MAX(Temp)
FROM       Weather
GROUP BY   Day(Time) AS day,
           Nation(Latitude, Longitude) AS nation;
```



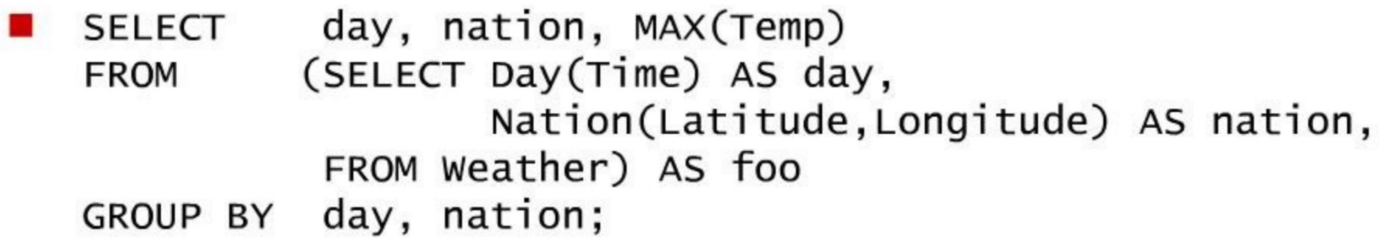
Problems with GROUP BY - Histograms

- The following is not a STANDARD SQL query!



```
SELECT    day, nation, MAX(Temp)
FROM      Weather
GROUP BY  Day(Time) AS day,
          Nation(Latitude, Longitude) AS nation;
```

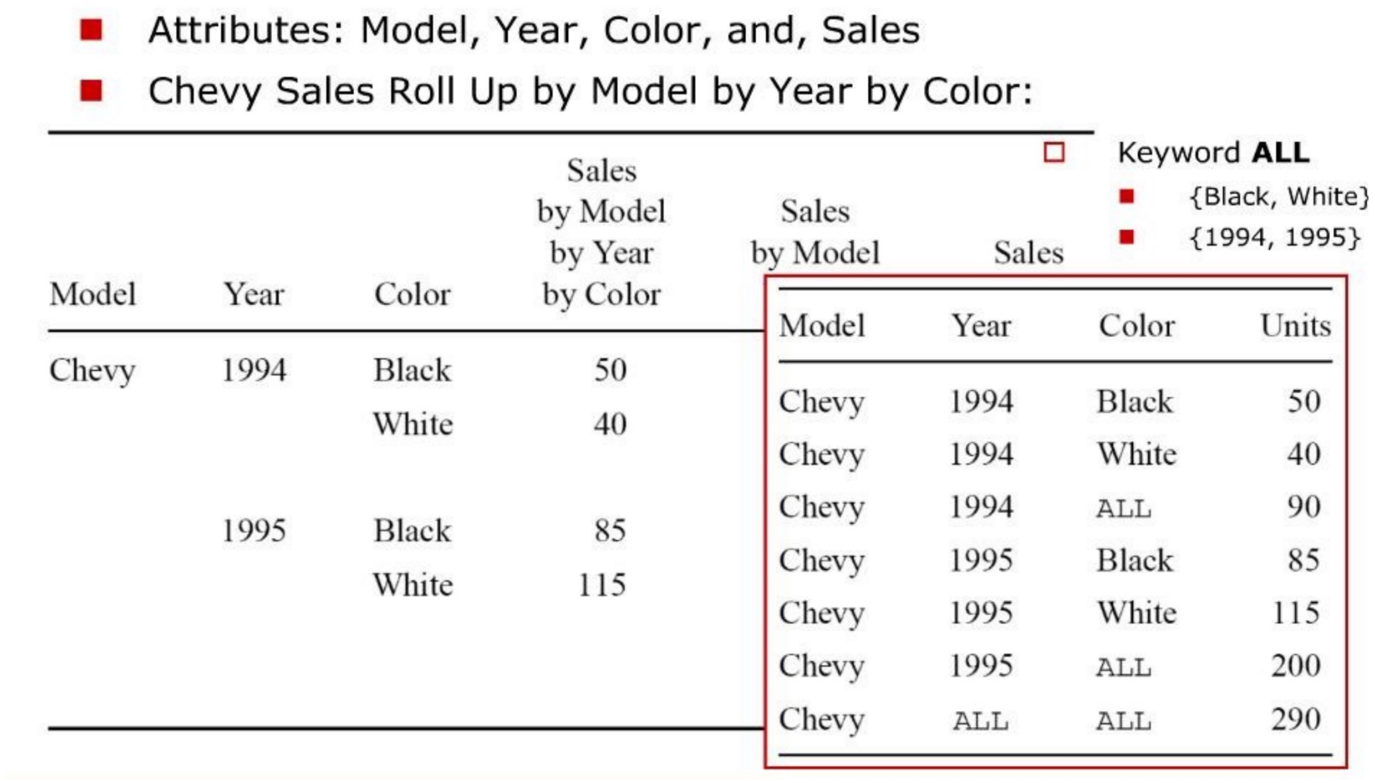
- In standard SQL:



```
SELECT    day, nation, MAX(Temp)
FROM     (SELECT Day(Time) AS day,
                 Nation(Latitude,Longitude) AS nation,
          FROM Weather) AS foo
GROUP BY  day, nation;
```

Problems with GROUP BY – Roll-Up Reports

- Users want roll-Up reports



- Attributes: Model, Year, Color, and, Sales
- Chevy Sales Roll Up by Model by Year by Color:

Model	Year	Color	Sales by Model by Year by Color	Sales by Model	Sales
Chevy	1994	Black	50		
		White	40		
	1995	Black	85		
		White	115		

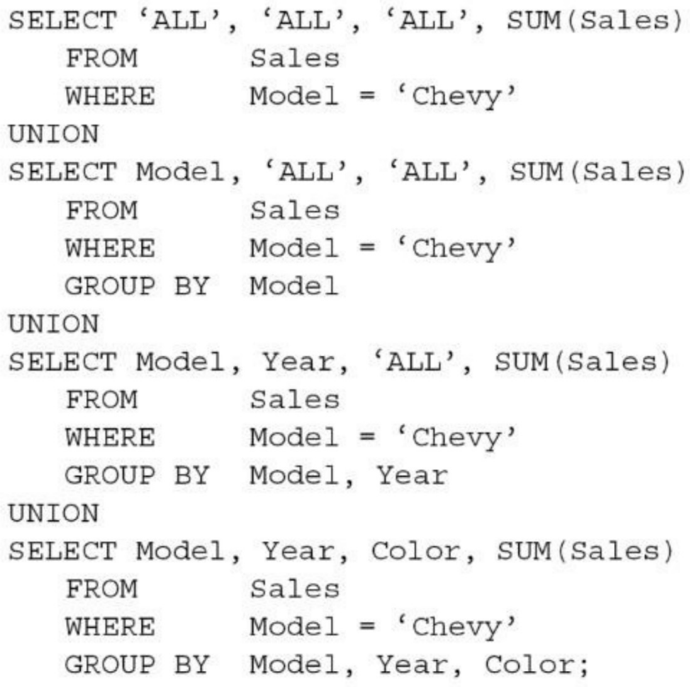
□ Keyword **ALL**

- {Black, White}
- {1994, 1995}

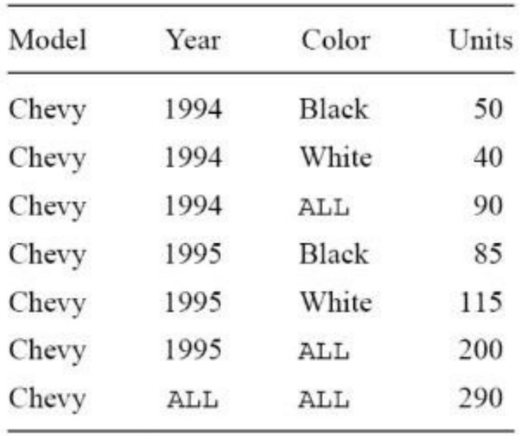
Model	Year	Color	Units
Chevy	1994	Black	50
Chevy	1994	White	40
Chevy	1994	ALL	90
Chevy	1995	Black	85
Chevy	1995	White	115
Chevy	1995	ALL	200
Chevy	ALL	ALL	290

Problems with GROUP BY – Roll-Up Reports

- To build the “Chevy Sales Roll Up”



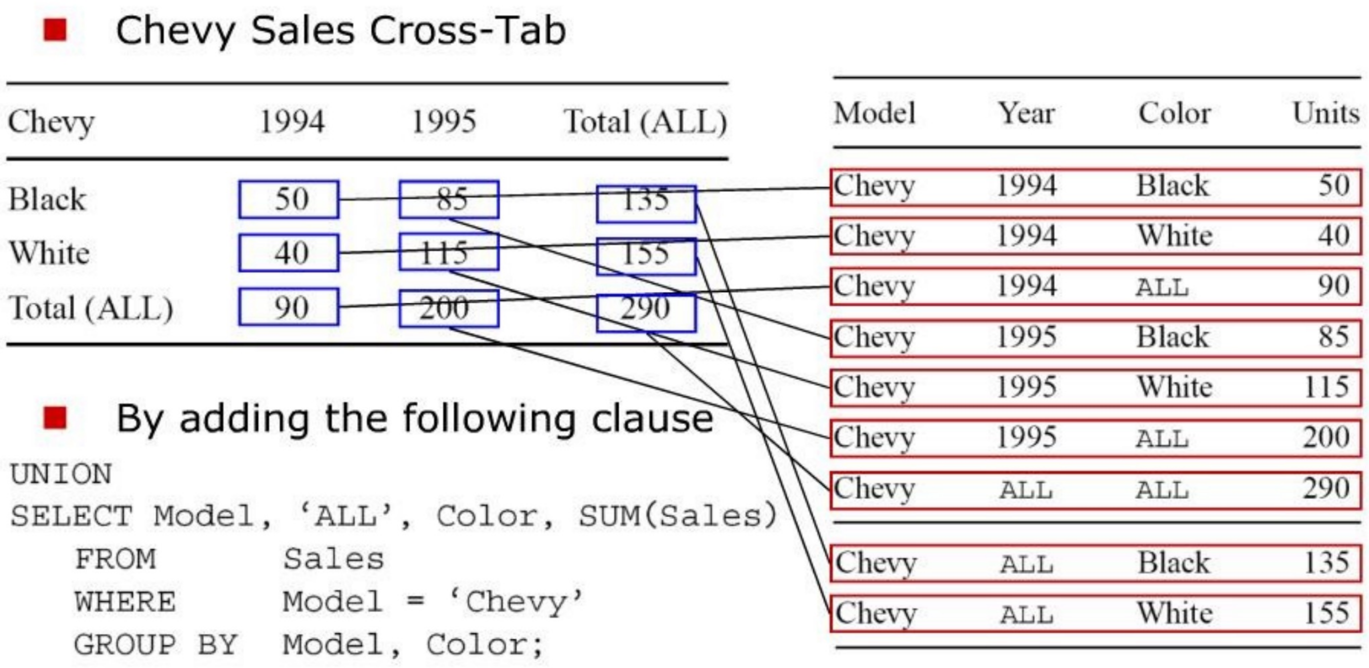
```
SELECT ‘ALL’, ‘ALL’, ‘ALL’, SUM(Sales)
    FROM       Sales
    WHERE      Model = ‘Chevy’
UNION
SELECT Model, ‘ALL’, ‘ALL’, SUM(Sales)
    FROM       Sales
    WHERE      Model = ‘Chevy’
    GROUP BY   Model
UNION
SELECT Model, Year, ‘ALL’, SUM(Sales)
    FROM       Sales
    WHERE      Model = ‘Chevy’
    GROUP BY   Model, Year
UNION
SELECT Model, Year, Color, SUM(Sales)
    FROM       Sales
    WHERE      Model = ‘Chevy’
    GROUP BY   Model, Year, Color;
```



Model	Year	Color	Units
Chevy	1994	Black	50
Chevy	1994	White	40
Chevy	1994	ALL	90
Chevy	1995	Black	85
Chevy	1995	White	115
Chevy	1995	ALL	200
Chevy	ALL	ALL	290

Problems with GROUP BY – Cross-Tabs

- Users want Cross-Tabulations



- Chevy Sales Cross-Tab

Chevy	1994	1995	Total (ALL)
Black	50	85	135
White	40	115	155
Total (ALL)	90	200	290

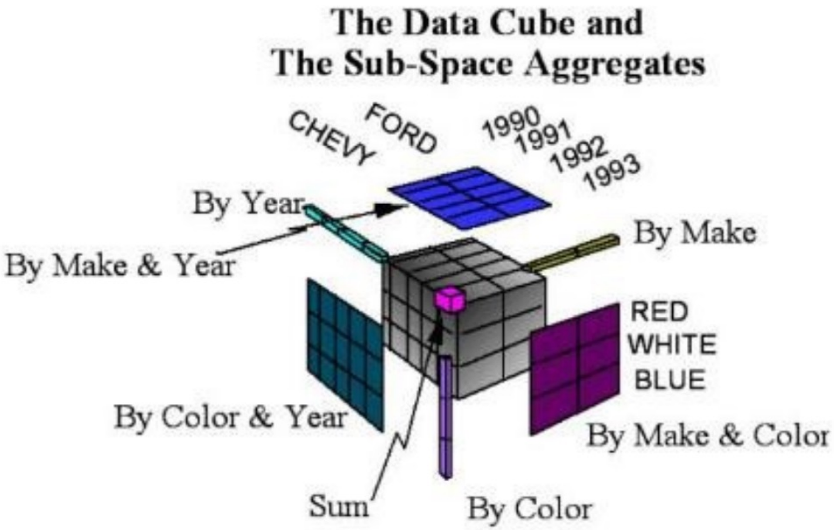
Model	Year	Color	Units
Chevy	1994	Black	50
Chevy	1994	White	40
Chevy	1994	ALL	90
Chevy	1995	Black	85
Chevy	1995	White	115
Chevy	1995	ALL	200
Chevy	ALL	ALL	290
Chevy	ALL	Black	135
Chevy	ALL	White	155

- By adding the following clause

```
UNION
SELECT Model, ‘ALL’, Color, SUM(Sales)
   FROM       Sales
   WHERE      Model = ‘Chevy’
   GROUP BY   Model, Color;
```

CUBE Operator

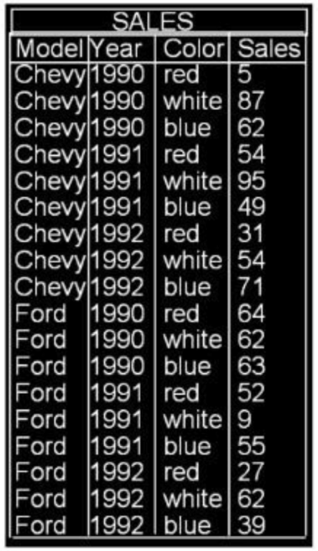
- Problems with GROUP BY
  - GROUP BY cannot directly construct:
    - Histograms
    - Roll-Up reports
    - Cross-Tabs
- CUBE Operator
  - Generalize GROUP BY and ROLL-UPP and Cross-Tabs.



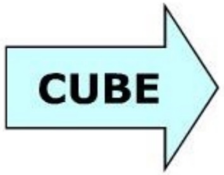
CUBE Operator

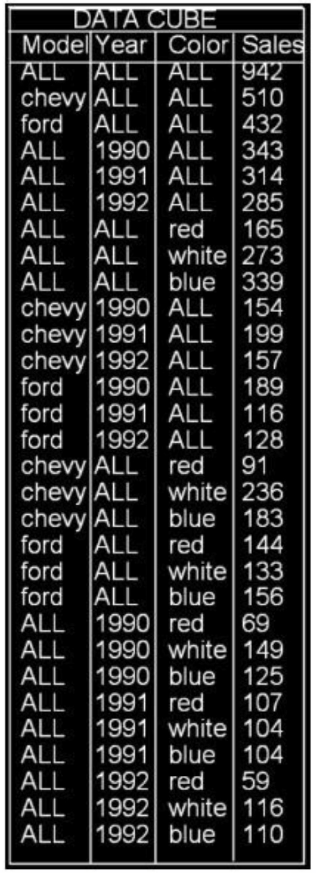
- Think of the N-dimensional Cube
- N-dimensional Aggregate [sum(), max(), ...]
  - Fits relational model exactly:
    - a_1, a_2, ..., a_n, f()
- Super-aggregate over N-1 dimensional sub-cubes
    - ALL, a_2, ..., a_n, f()
    - a_1, ALL, a_3, ..., a_n, f()
    - ...
    - a_1, a_2, ..., ALL, f()
  - This is the N-1 dimensional cross-tab.
- Super-aggregate over N-2 dimensional sub-cubes
    - ALL, ALL, a_3, ..., a_n, f()
    - ...
    - a_1, a_2, ..., ALL, ALL, f()
- ...

CUBE Operator – An example



SALES			
Model	Year	Color	Sales
Chevy	1990	red	5
Chevy	1990	white	87
Chevy	1990	blue	62
Chevy	1991	red	54
Chevy	1991	white	95
Chevy	1991	blue	49
Chevy	1992	red	31
Chevy	1992	white	54
Chevy	1992	blue	71
Ford	1990	red	64
Ford	1990	white	62
Ford	1990	blue	63
Ford	1991	red	52
Ford	1991	white	9
Ford	1991	blue	55
Ford	1992	red	27
Ford	1992	white	62
Ford	1992	blue	39

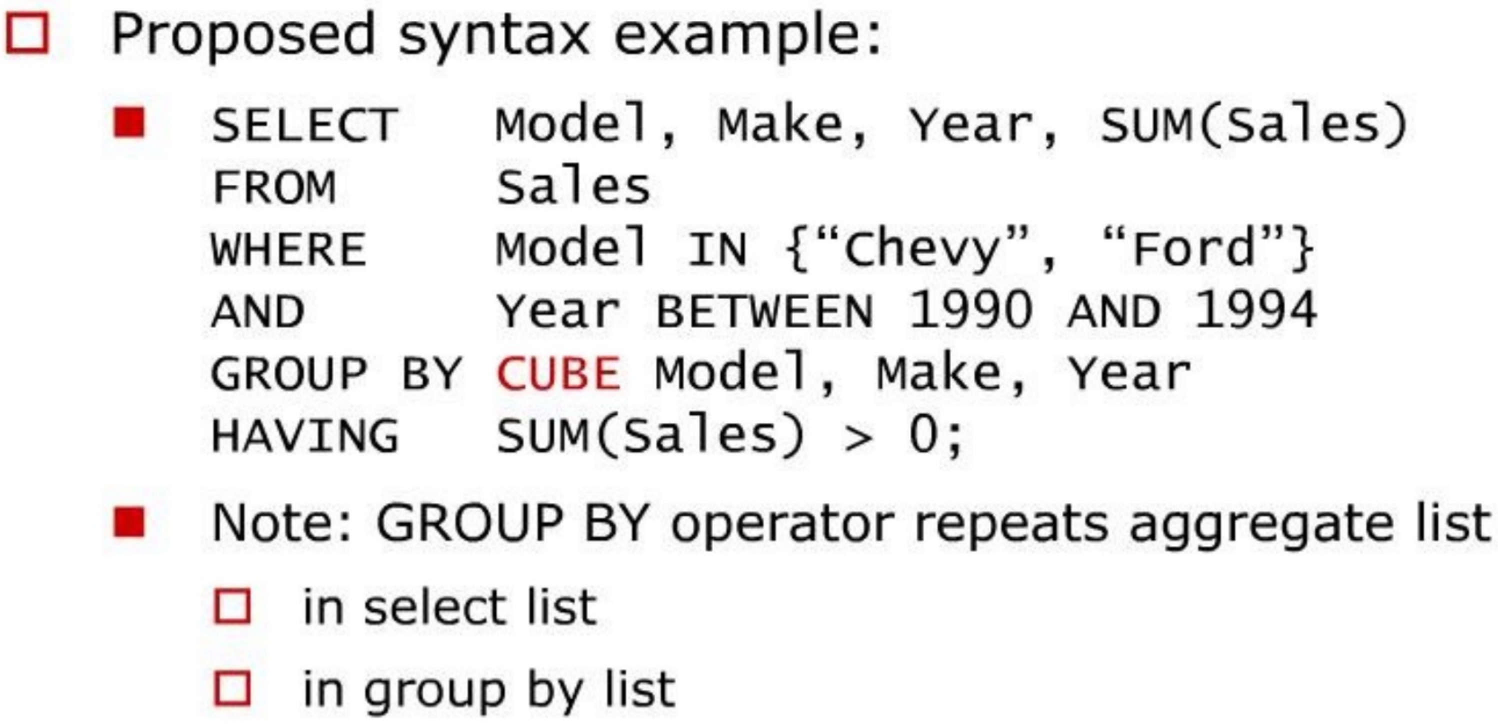




DATA CUBE			
Model	Year	Color	Sales
ALL	ALL	ALL	942
chevy	ALL	ALL	510
ford	ALL	ALL	432
ALL	1990	ALL	343
ALL	1991	ALL	314
ALL	1992	ALL	285
ALL	ALL	red	165
ALL	ALL	white	273
ALL	ALL	blue	339
chevy	1990	ALL	154
chevy	1991	ALL	199
chevy	1992	ALL	157
ford	1990	ALL	189
ford	1991	ALL	116
ford	1992	ALL	128
chevy	ALL	red	91
chevy	ALL	white	236
chevy	ALL	blue	183
ford	ALL	red	144
ford	ALL	white	133
ford	ALL	blue	156
ALL	1990	red	69
ALL	1990	white	149
ALL	1990	blue	125
ALL	1991	red	107
ALL	1991	white	104
ALL	1991	blue	104
ALL	1992	red	59
ALL	1992	white	116
ALL	1992	blue	110

- Think of ALL as a token representing the set:
  - {red, white, blue}
  - {1990, 1991, 1992}
  - {Chevy, Ford}

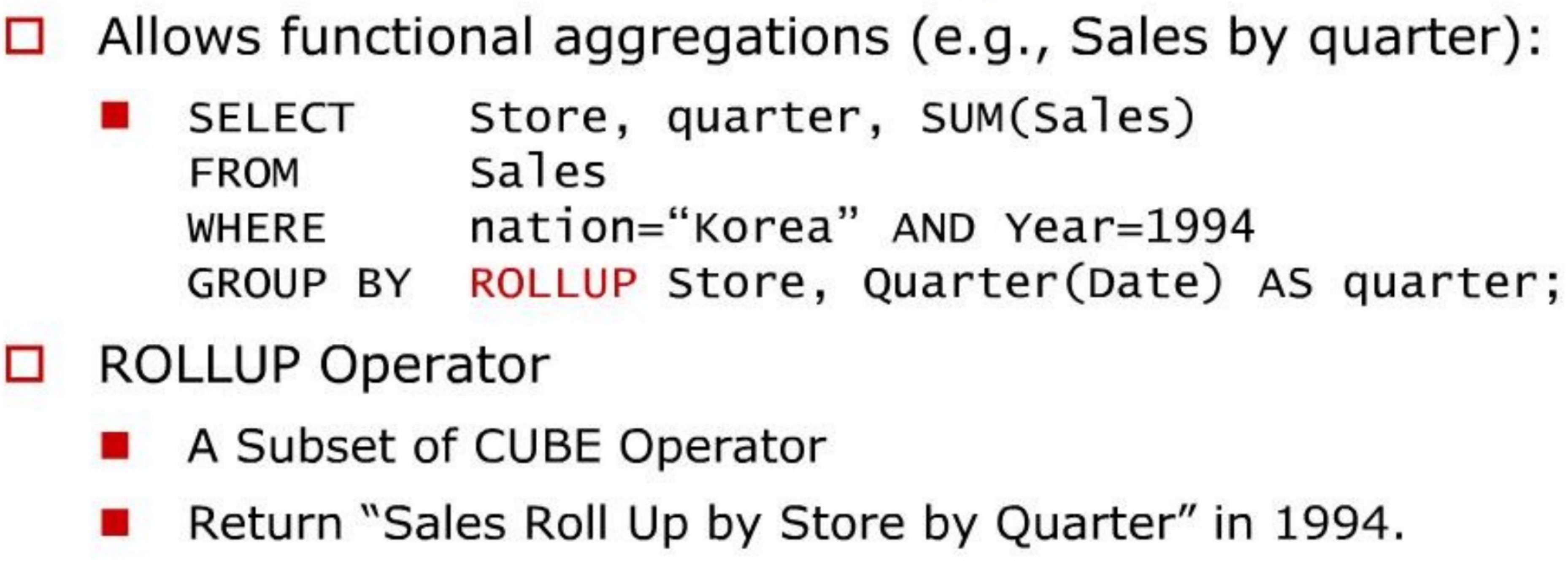
CUBE Operator – Syntax



- Proposed syntax example:

  -
    ```
    SELECT    Model, Make, Year, SUM(Sales)
    FROM      Sales
    WHERE     Model IN {“Chevy”, “Ford”}
    AND       Year BETWEEN 1990 AND 1994
    GROUP BY CUBE Model, Make, Year
    HAVING    SUM(Sales) > 0;
    ```
  - Note: GROUP BY operator repeats aggregate list
    - in select list
    - in group by list

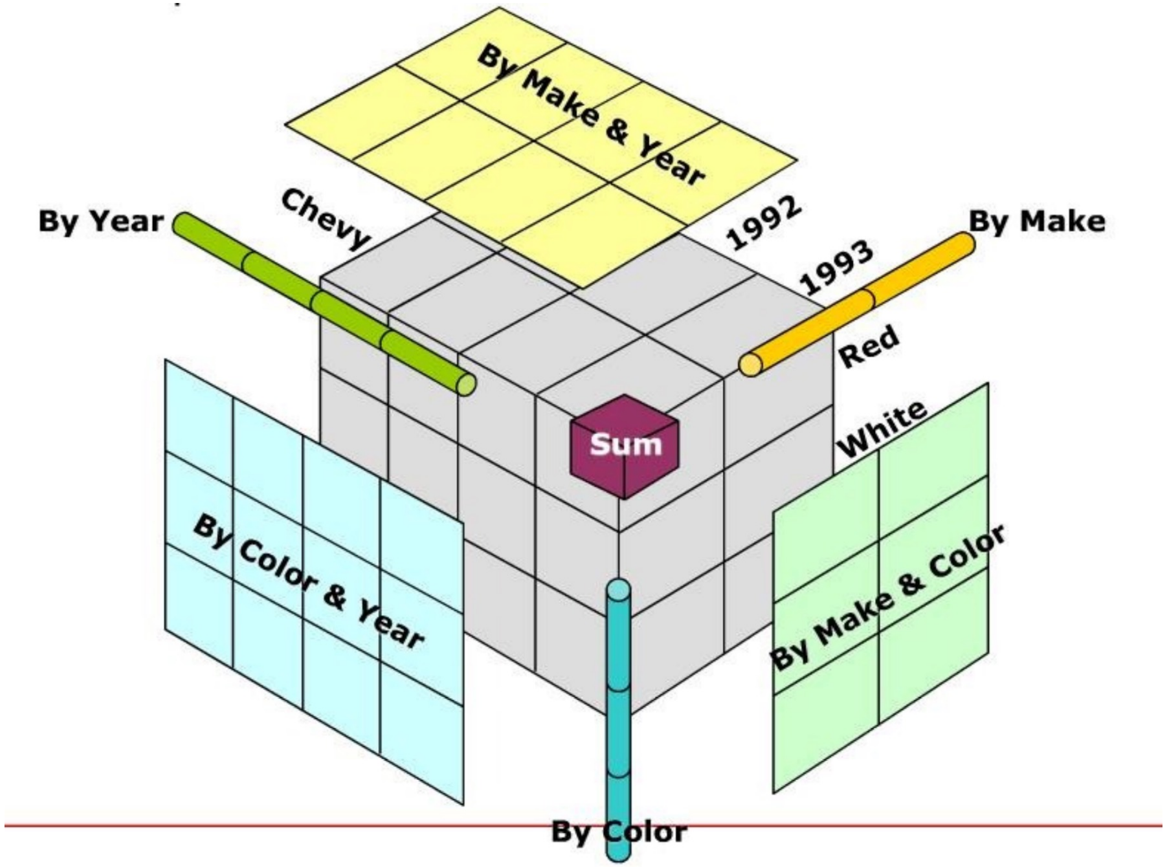
CUBE Operator – Functional Aggregation



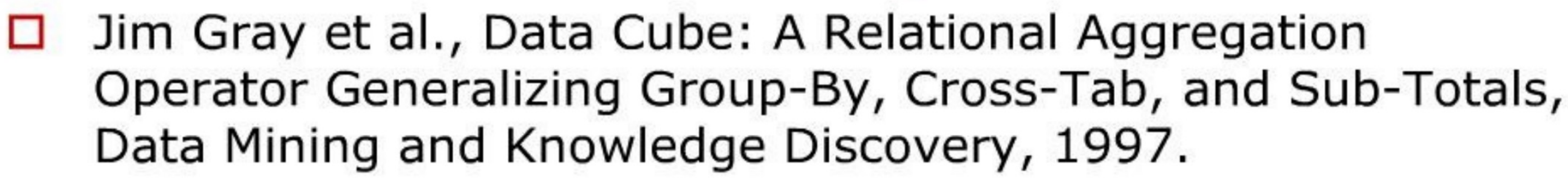
- Allows functional aggregations (e.g., Sales by quarter):

  -
    ```
    SELECT     Store, quarter, SUM(Sales)
    FROM       Sales
    WHERE      nation="Korea" AND Year=1994
    GROUP BY   ROLLUP Store, Quarter(Date) AS quarter;
    ```
- ROLLUP Operator
  - A Subset of CUBE Operator
  - Return "Sales Roll Up by Store by Quarter" in 1994.

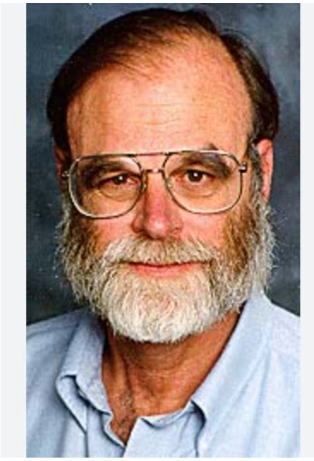
CUBE Operator – An Example of 3D Data Cube



References



- □ Jim Gray et al., Data Cube: A Relational Aggregation Operator Generalizing Group-By, Cross-Tab, and Sub-Totals, Data Mining and Knowledge Discovery, 1997.



Microsoft Researcher Jim Gray Receives Turing Award for Helping to Transform Databases into Dynamic Tools Used by Millions