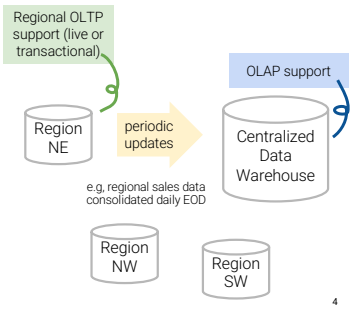
Data Science Initiative
BROWN

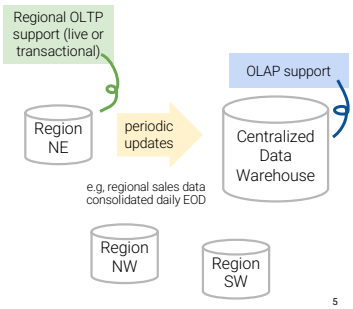
Post-hoc large-scale analysis (**OLAP**) happens separately from updates (**OLTP**).

- **OLAP** is performed in a data warehouse
- **OLTP** happens in the per-region database
- OLAP **separate from critical path** of OLTP.



Data Science Initiative
BROWN

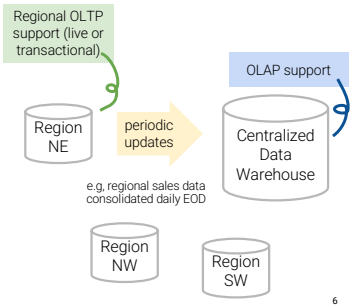
- **Extract** useful business info to be summarized
- **Transform** it (i.e., canonicalize/clean up)
- **Load** it into the warehouse



Data Science Initiative
BROWN

- **Extract** useful business info to be summarized
- **Transform** it (i.e., canonicalize/clean up)
- **Load** it into the warehouse

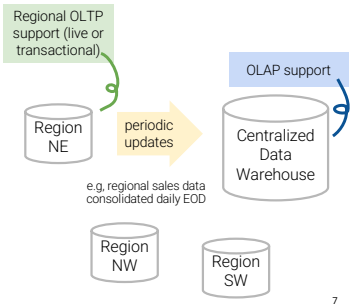
Note: **OLAP will suffer some staleness**, but OK in practice



Data Science Initiative
BROWN

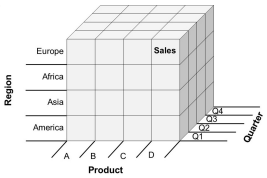
- **Extract** useful business info to be summarized
- **Transform** it (i.e., canonicalize/clean up)
- **Load** it into the warehouse

Note: **OLAP will suffer some staleness**, but OK in practice



Data Science Initiative
BROWN

Multi-dimensional Data Aka Data Cube

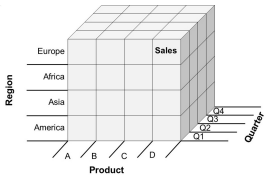


OLAP systems fundamentally provide comprehensive, summarized views of data in data warehouses.

This summarization can and will happen across multiple dimensions of data. One such representation is a data cube, aka OLAP cube.

Data Science Initiative
BROWN

Multi-dimensional Data Aka Data Cube



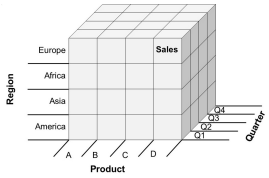
Roll-up and Drill-down

Cross-tab

Slice and Dice

Data Science Initiative
BROWN

- We will look at OLAP operators using the data cube below and the data on the right.
- For simplicity, let's assume only 2 products and 2 regions



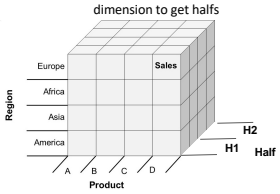
PRODUCT	QUARTER	REGION	SALES
A	Q1	Europe	10
A	Q1	America	20
A	Q2	Europe	20
A	Q2	America	50
A	Q3	America	20
A	Q4	Europe	10
A	Q4	America	30
B	Q1	Europe	40
B	Q1	America	60
B	Q2	Europe	20
B	Q2	America	10
B	Q3	America	20
B	Q4	Europe	10
B	Q4	America	40

Data Science Initiative
BROWN

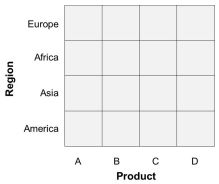
Roll-up

The operation of adjusting granularity to summarize at different levels of a dimension hierarchy

Rolling-up across the time dimension to get halfs



Rolling-up across the time dimension entirely



Data Science Initiative
BROWN

Roll-up

The operation of adjusting granularity to summarize at different levels of a dimension hierarchy

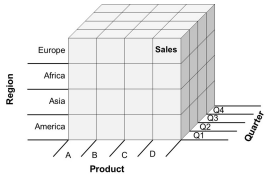
Rolling-up across the region and time dimensions

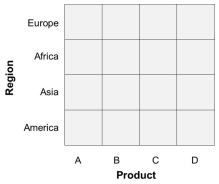
Data Science Initiative
BROWN

Cross-tab

Aka pivot-table - aggregations of groups of individual values from a more extensive table within one or more discrete categories

Example: number of sales per **region** per **product**



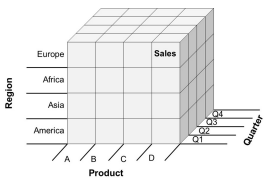


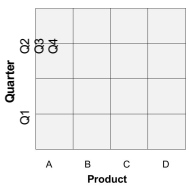
Data Science Initiative
BROWN

Cross-tab

Aka pivot-table - aggregations of groups of individual values from a more extensive table within one or more discrete categories

Example: number of sales per **quarter** per **product**



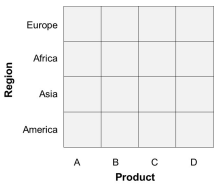


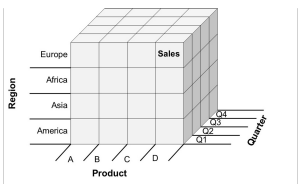
Data Science Initiative
BROWN

Drill-down

The operation of adjusting granularity to summarize at different levels of a dimension hierarchy

Drill-down time dimension to quarters



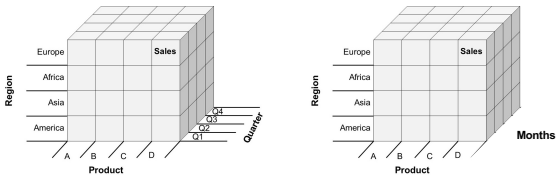


Data Science Initiative
BROWN

Drill-down

The operation of adjusting granularity to summarize at different levels of a dimension hierarchy

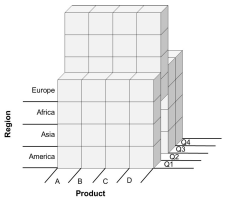
Drill-down time dimension to months?



Data Science Initiative
BROWN

Slicing

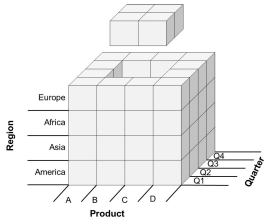
An operation in which one of the dimension is set to a particular value



Data Science Initiative
BROWN

Dicing

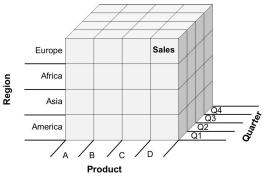
An operation consisting in a range selection on one or more dimensions



Data Science Initiative
BROWN

How do we actually store our data cube?

- **Dense tensors** are not efficient storage.
  - Many elements are likely nulls, i.e., not measured!



Data Science Initiative
BROWN

How do we actually store our data cube?

- Normalized data in long format
- Star Schema

OLAP in a relational database:

- **mental model**: data cube, a multidimensional tensor
- **physical model**: star schema or snowflake schema of relations

PRODUCT	QUARTER	REGION	SALES
A	Q1	Europe	10
A	Q1	America	20
A	Q2	Europe	20
A	Q2	America	50
A	Q3	America	20
A	Q4	Europe	10
A	Q4	America	30
B	Q1	Europe	40
B	Q1	America	60
B	Q2	Europe	20
B	Q2	America	10
B	Q3	America	20
B	Q4	Europe	10
B	Q4	America	40

Data Science Initiative
BROWN

- SQL-99 introduced 3 extensions to the GROUP BY statement: the CUBE, ROLLUP and GROUPING SETS operator
- The **CUBE** operator computes a union of GROUP BY's on every subset of the specified attribute types

Data Science Initiative BROWN

Group by CUBE

```
SELECT quarter, region, SUM(sales)
FROM sales_table
GROUP BY CUBE (quarter, region)
```

- this query computes the union of the following groupings of the `sales_table` being: {(quarter,region), (quarter), (region), ()}, where () denotes an empty group list
- resulting multiset will have how many tuples?

Data Science Initiative BROWN

Group by CUBE

```
SELECT quarter, region, SUM(sales)
FROM sales_table
GROUP BY CUBE (quarter, region)
```

- this query computes the union of the following groupings of the `sales_table` being: {(quarter,region), (quarter), (region), ()}, where () denotes an empty group list
- resulting multiset will have:
  - 4*2+4*1+1*2+1 or 15 tuples

Data Science Initiative BROWN

GROUP BY CUBE query

PRODUCT	QUARTER	REGION	SALES
A	Q1	Europe	10
A	Q1	America	20
A	Q2	Europe	20
A	Q2	America	50
A	Q3	America	20
A	Q4	Europe	10
A	Q4	America	30
B	Q1	Europe	40
B	Q1	America	60
B	Q2	Europe	20
B	Q2	America	10
B	Q3	America	20
B	Q4	Europe	10
B	Q4	America	40

```
SELECT QUARTER,
REGION, SUM(SALES)
FROM SALESTABLE
GROUP BY CUBE
(QUARTER, REGION)
```

QUARTER	REGION	SALES
Q1	Europe	50
Q1	America	80
Q2	Europe	40
Q2	America	60
Q3	Europe	NULL
Q3	America	40
Q4	Europe	20
Q4	America	80
Q1	NULL	130
Q2	NULL	100
Q3	NULL	40
Q4	NULL	90
NULL	Europe	110
NULL	America	250
NULL	NULL	360

Total for Q1 for all regions

Total for Europe

Total for America

Total for all regions

Data Science Initiative BROWN

GROUP BY ROLLUP Query

PRODUCT	QUARTER	REGION	SALES
A	Q1	Europe	10
A	Q1	America	20
A	Q2	Europe	20
A	Q2	America	50
A	Q3	America	20
A	Q4	Europe	10
A	Q4	America	30
B	Q1	Europe	40
B	Q1	America	60
B	Q2	Europe	20
B	Q2	America	10
B	Q3	America	20
B	Q4	Europe	10
B	Q4	America	40

```
SELECT QUARTER, REGION,
SUM(SALES)
FROM SALESTABLE
GROUP BY ROLLUP (QUARTER,
REGION)
```

The primary rollup dimension is quarter unlike in CUBE query

QUARTER	REGION	SALES
Q1	Europe	50
Q1	America	80
Q2	Europe	40
Q2	America	60
Q3	Europe	NULL
Q3	America	40
Q4	Europe	20
Q4	America	80
Q1	NULL	130
Q2	NULL	100
Q3	NULL	40
Q4	NULL	90
NULL	NULL	360

Total for Q1 for all regions

Total for all quarters for all regions

Data Science Initiative BROWN

Grouping Sets

```
SELECT quarter, region, SUM(sales)
FROM sales_table
GROUP BY GROUPING SETS (
    (quarter, region),
    (quarter),
    (region)
    ()
);
```

- this query computes the union of the following groupings of the `sales_table` being: {(quarter,region), (quarter), (region), ()}, where () denotes an empty group list

Data Science Initiative BROWN

Grouping Sets

```
SELECT quarter, region, SUM(sales)
FROM sales_table
GROUP BY GROUPING SETS (
    (quarter, region),
    (quarter)
);
```

- this query computes the union of the following groupings of the `sales_table` being: {(quarter,region), (quarter)}