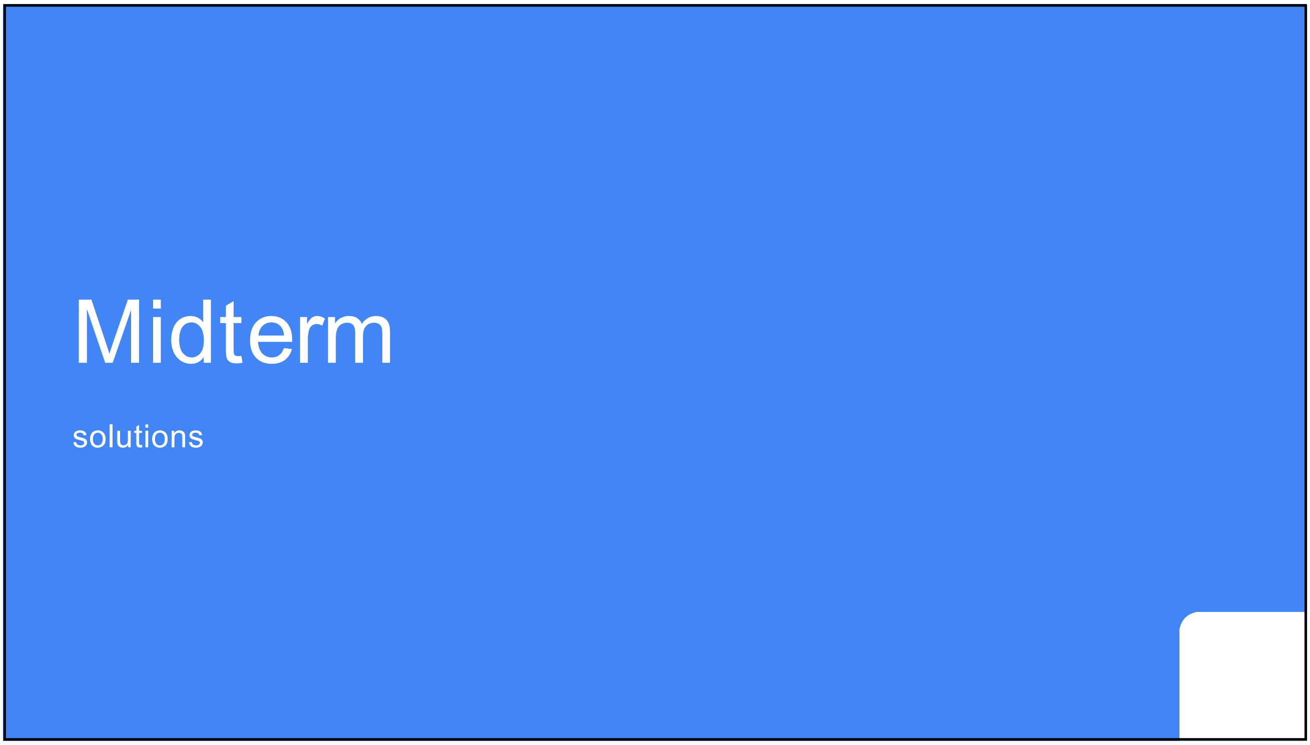
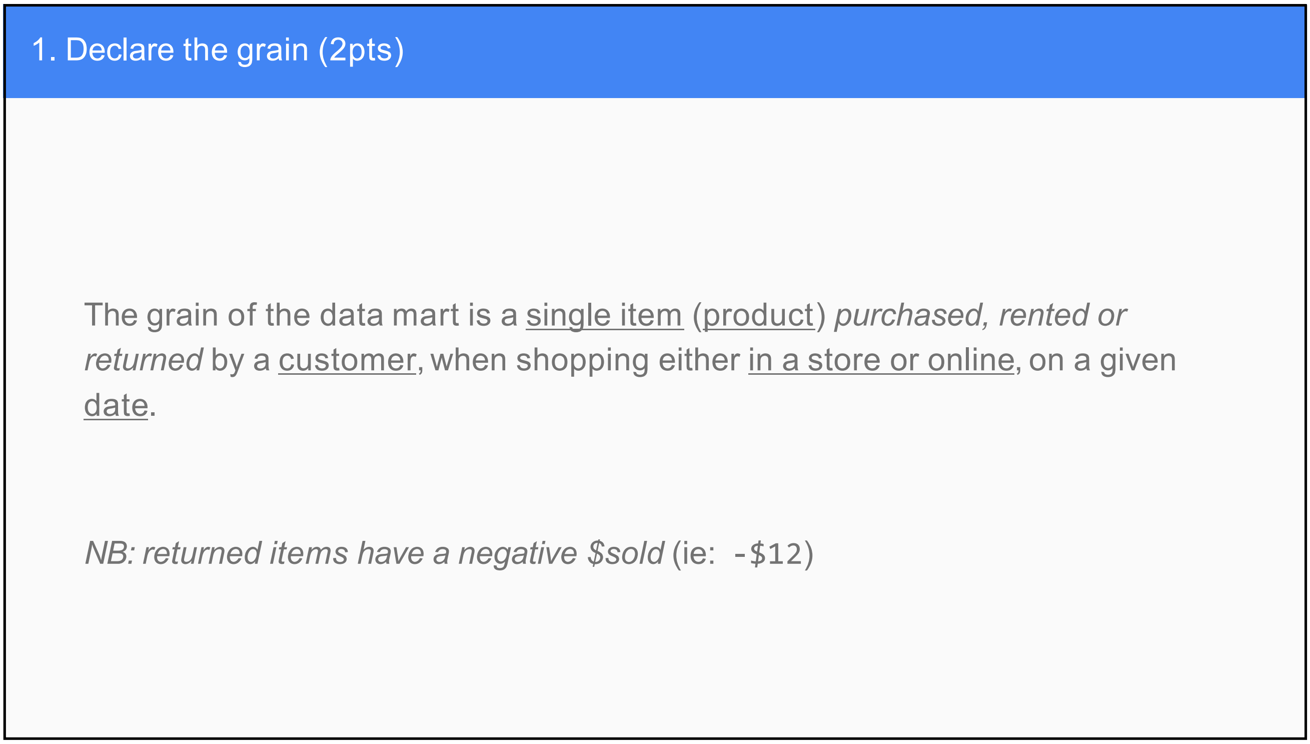


Midterm

solutions



1. Declare the grain (2pts)

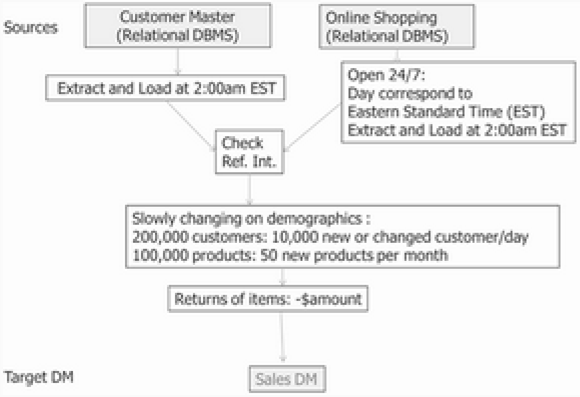
The grain of the data mart is a single item (product) *purchased, rented or returned* by a customer, when shopping either in a store or online, on a given date.

NB: returned items have a negative $sold (ie: `-$12`)

2. Steps to integrate two databases, with ref. to the high-level data staging plan (4pts)

Refer to slides on Data Staging.
Specifically, from slide 7:

- Create a very high-level, one-page schematic of the source-to-target flow
- Identify starting and ending points
- Label known data sources
- Include placeholders for sources yet to be determined
- Label targets
- Include notes about known problems



Sources

Customer Master
(Relational DBMS)

Online Shopping
(Relational DBMS)

Extract and Load at 2:00am EST

Open 24/7:
Day correspond to
Eastern Standard Time (EST)
Extract and Load at 2:00am EST

Check
Ref. Int.

Slowly changing on demographics :
200,000 customers: 10,000 new or changed customer/day
100,000 products: 50 new products per month

Returns of items: -$amount

Target DM

Sales DM

3. Draw the dimensional model (10pts)

~ “*classical*” sales data mart, modified for online sales

Fact table

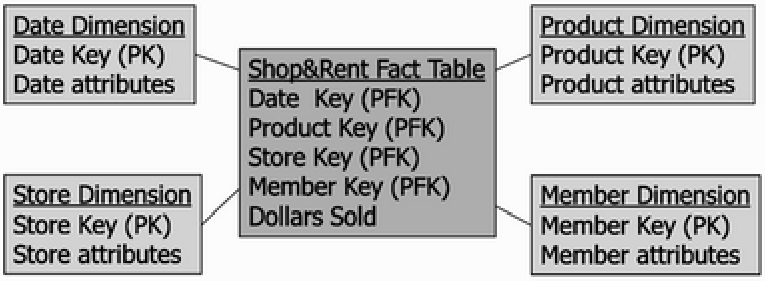
- Dollars sold refers to line item on a bill/invoice
- Can include role playing dates (sold/returned)
    - returned date used only (hopefully) for rentals

Product dimension

- Lists items for sale, includes rentable? ⚑

Store dimension

- 1 row for online with @home delivery (no location)
- All other stores have an online? ⚑



Date Dimension
Date Key (PK)
Date attributes

Shop&Rent Fact Table
Date Key (PFK)
Product Key (PFK)
Store Key (PFK)
Member Key (PFK)
Dollars Sold

Product Dimension
Product Key (PK)
Product attributes

Store Dimension
Store Key (PK)
Store attributes

Member Dimension
Member Key (PK)
Member attributes

4. Example of aggregate/ cube to speed up queries (a & b) (2pts)

- How does the total sales of a product during the current month (e.g. February 2018) compare to the sales during the Summer of 2017?
- How does the total sales of a product during this month (e.g. February 2018) compare to the sales at the same time (e.g. February 2017) last year?

An aggregate by Month would speed both queries up.

Here, we use the Year Month Day concept hierarchy.

5. Show how a star join operation can optimize query (c) (4pts)

What are the names and brands of the six tents that had the lowest volume of sales, in Ontario, during the week starting on 25 June 2017? That is, we want to determine which tents did not sell well during the so-called "peak" summer selling season.

Refer to Slide 27 of the Physical Design deck.

1. **Product dimension**: select tents, then semi-join to fact table to get fact IDs
2. **Date dimension**: select rows for that week, semi-join to fact table to get fact IDs
3. Compute intersection of the above 2 reduced dimensions
4. Group by product and compute count *of facts for that product*
5. Sort results by lowest count
6. List the names and brands of the 6 with the lowest counts.

⚑ This query is based on the idea of selectivity. Intuitively, we will have fewer tents. Also, assume we have the data of 10 years. In this case, retrieving the data of one week will reduce the workload considerably, especially if our data are partitioned (on disk or in the cloud) by date

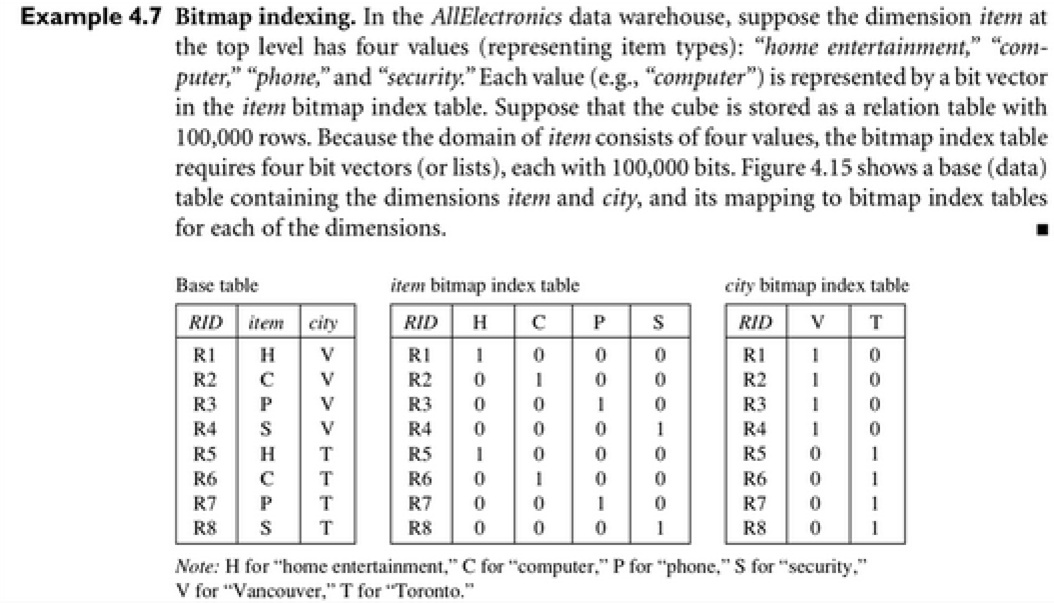
6. Show how a bitmap index is used to optimize query (d) (2pts)

~ what colour was the best selling women's hat in 2017?

- Maintain a bitmap for colours [red, black, lilac, green]
- Join bitmap with fact table
- Most popular hat colour has the most # '1's in bitmap

Red	**100000001000...**
Black	**011011000111...**
Lilac	**000100000000...**
Green	**000000110000...**

Bitmap index



Example 4.7 Bitmap indexing. In the *AllElectronics* data warehouse, suppose the dimension *item* at the top level has four values (representing item types): "*home entertainment,*" "*computer,*" "*phone,*" and "*security.*" Each value (e.g., "*computer*") is represented by a bit vector in the *item* bitmap index table. Suppose that the cube is stored as a relation table with 100,000 rows. Because the domain of *item* consists of four values, the bitmap index table requires four bit vectors (or lists), each with 100,000 bits. Figure 4.15 shows a base (data) table containing the dimensions *item* and *city*, and its mapping to bitmap index tables for each of the dimensions. ■

Base table

RID	*item*	*city*
R1	H	V
R2	C	V
R3	P	V
R4	S	V
R5	H	T
R6	C	T
R7	P	T
R8	S	T

item bitmap index table

RID	H	C	P	S
R1	1	0	0	0
R2	0	1	0	0
R3	0	0	1	0
R4	0	0	0	1
R5	1	0	0	0
R6	0	1	0	0
R7	0	0	1	0
R8	0	0	0	1

city bitmap index table

RID	V	T
R1	1	0
R2	1	0
R3	1	0
R4	1	0
R5	0	1
R6	0	1
R7	0	1
R8	0	1

Note: H for "home entertainment," C for "computer," P for "phone," S for "security,"
V for "Vancouver," T for "Toronto."

7. Provide the SQL statement to answer query (e). (4pts)

Name the five (5) bicycles are the most popular in the Ottawa store, in terms of the total number of sales in 2017, when compared to the sales of bicycles throughout Canada.

For instance, the Ghost Trekking 5 bicycle was the most popular seller in the Ottawa store, during 2017, However, it was ranked 3rd popular in Canada during the same period of time.

This is an example of an iceberg query

```
SELECT P.pname
FROM Fact S, Products P, Date D, Customer C
WHERE S.pid = P.pid AND S.did = D.did AND S.cid = C.cid
AND   C.city = `Ottawa`
AND  D.year = 2017
AND P.type = `bike`
GROUP BY P.name
ORDER BY count(*) DESC
OPTIMIZE FOR 5 ROWS (or limit 5)
```

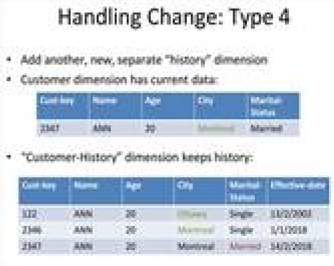
same idea for Canada wide, but replace City by Country=Canada

8. How do you handle a slowly-changing-dimension (SCD) on marital status? (6pts)

The answer will depend on whether we want to maintain a link to prior marital status(es). Refer to the Data Staging slides 54 to 60.

SCD Type:

1. **overwrite the old value** by the new one, but this is *not recommended*!
2. **add a new row.** New values referring to this person uses the new ID, existing data will use old ID.
   a. **add a "flag"** is row current
   b. **add an "effective date" attribute**
3. **add a new attribute,** where we have "old marital status" as well as "new marital status", with an effective date
4. **add a new "History" dimension** (see example)



Handling Change: Type 4

- Add another, new, separate "history" dimension
- Customer dimension has current data:

Cust key	Name	Age	City	Marital Status
2347	ANN	20	Montreal	Married

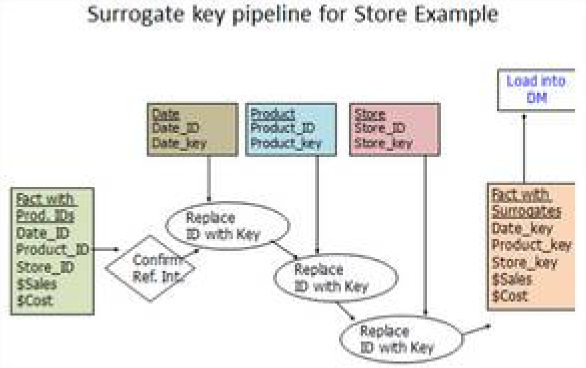
- "Customer-History" dimension keeps history:

Cust key	Name	Age	City	Marital Status	Effective date
122	ANN	20	Ottawa	Single	13/2/2003
2346	ANN	20	Montreal	Single	1/1/2018
2347	ANN	20	Montreal	Married	14/2/2018

9. Explain what is the surrogate key pipeline (2pts)

Refers to the data staging steps we take to convert the Fact table from the transactional input data format to the dimensional model.

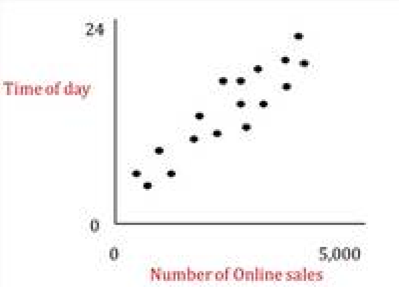
Refer to slide 26 of the Data Staging nodes, as well as the textbook by Kimball et. al.

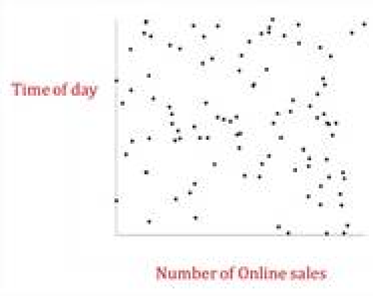


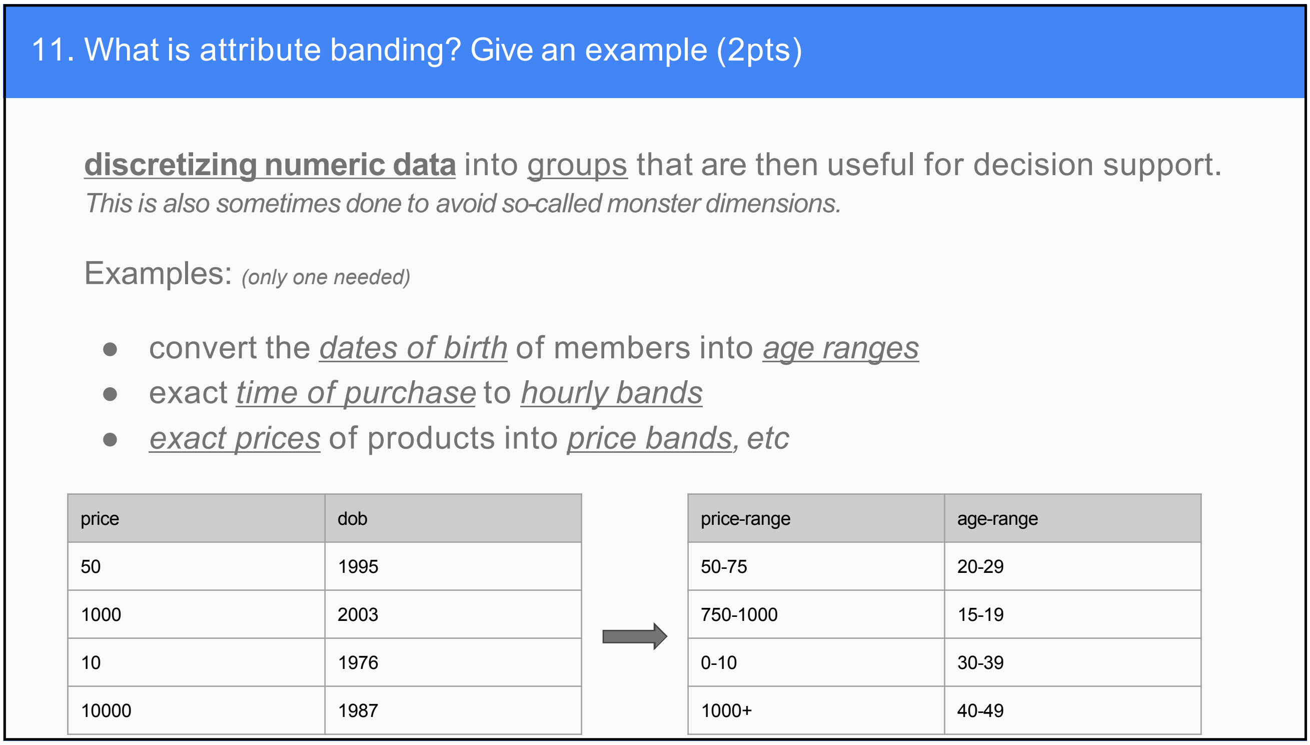
10. How to determine correlation [time of the day/ number of online sales] (2pts)

Different strategies, including:

- Statistical tests
  - Pearson's Coefficient (-1 or 1 indicates correlation, 0 indicates none)
- Drawing scatter plots







11. What is attribute banding? Give an example (2pts)

discretizing numeric data into groups that are then useful for decision support.
This is also sometimes done to avoid so-called monster dimensions.

Examples: *(only one needed)*

- convert the *dates of birth* of members into *age ranges*
- exact *time of purchase* to *hourly bands*
- *exact prices* of products into *price bands, etc*

price	dob
50	1995
1000	2003
10	1976
10000	1987

→

price-range	age-range
50-75	20-29
750-1000	15-19
0-10	30-39
1000+	40-49