**Dimensional Modeling: Guided Reading Questions**

Ross, M., and R. Kimball, *The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling*, 3rd Edition, John Wiley and Sons, 2013.

access as a Safari book online *Miami Libraries, Databases A-Z*. See instructions in a separate document.

Reading assignment: Ch. 1 sections up to “Kimball’s DW/BI Architecture” (stop here)

Skip (no need to read):

* + “Publishing Metaphor for DW/BI Managers”
  + “OLAP Deployment Considerations”

1. What are the two purposes of an organization’s information?

Operational record keeping and analytical decision making.

1. How does this relate to what kind of data is processed by operational systems and DW/BI systems?

Operational Systems(Input) update current data and do not maintain historical data. DW/BI (Output) on the other hand deal with a large amount of detailed current and historical data.

1. Which system uses current data? For what purpose?

Operational Systems, in order to update current records and maintain the most current state of information

1. Which system uses historical data? For what purpose?

DW/BI in order to perform widespread data analytics over a large period of time.

1. Describe the scenario that the authors call an “imposter” DW/BI system.

For some companies, their “DW/BI” systems are a copy of their operational systems on a different hardware platform.

1. Why are BI queries considered “unpredictable?”

With so many hundreds of tables to keep track of, a simple user query cannot possibly build a query to retrieve the exact correct data. Attempting monster queries overwhelms the optimizers which yields results that are not consistent or desirable.

1. Why are BI queries considered “complex?”

BI systems are very optimized and very normalized for storing a large swath of data. As a result, queries that retrieve the actual desired results often require including a huge number of columns across a huge number of tables.

1. What are the two main requirements addressed by dimensional modeling?

User understandability and query performance (also provides resilience to change)

1. What are the main differences between how an operational database and a data warehouse structure data?

Operational Databases utilize highly normalized structures which requires distributed a single table into several other tables. Data warehouses prefer to utilize dimensional (de-normalized) models which apply concrete rules to each “layer” of data where the number of layers is indeterminate.

1. What characteristics describe a measure attribute stored in a fact table?

Measure attributes should be the lowest-level of measurement data and they can not be replicated elsewhere, numeric.

1. Define each of these terms and give an example of a numeric measure (not from the chapter) that is
   1. Additive

Able to be summed to a valid total in any dimension. Ex: Cost of items on a grocery list.

* 1. semi-additive

Can be added together, but not when placed under certain constraints (like time).

If you wanted to see loans taken out on specific days

* 1. non-additive

Values that cannot be combined due to them existing in separate contexts. A person’s net worth based on their possessions. (GPA and Unit Prices)

1. Why should text fields be avoided as measures?

Text takes up a larger amount of space, can usually be quantified into a discrete value, and can much more easily be analyzed if among other non-textual data.

1. Which kind of attributes are mostly used in query constraints and groupings?

Dimension Attributes

1. Which tables are typically larger, fact or dimension tables? Why?

Fact Tables, due to their massive number of rows to account for all the granular bits of information description.

1. Study the query below and identify which attributes are fact table attributes (measures) and which are dimension attributes:

select continent, year, sum(population), avg(life\_exp), min(income), max(income)

from world\_health\_data\_viz

where year >=1900

group by continent, year;

dimension = continent, year

fact table attributes = population, life\_exp, income