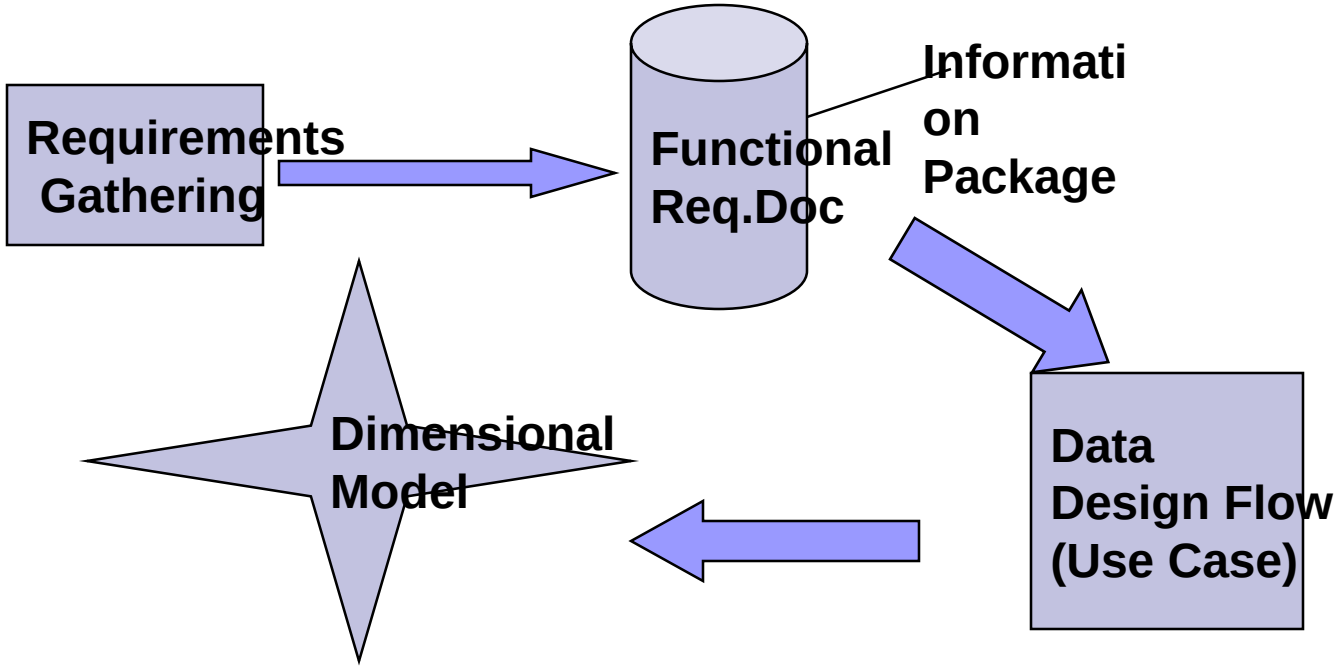
Dimensional Modeling Basic to Advance Design

Manoj kuamr, NIIT

From Requirement to Logical Design

- The requirements definition completely drives the Data design for the DW.
- Data design consists of putting together the data structures.
- A group of data elements form a data structure.
- Logical data design includes determination of the various data elements that are needed and combination of the data elements into structures of data and also establishing relationships among the data structures.



Dimensional Modeling Overview

- What is Dimensional Modeling
  - It is a logical design technique to structure the business dimensions and the metrics that are analyzed along these dimensions.
  - A logical design technique that seeks to present the data in a standard framework that is intuitive and allows for high performance access
  - It is inherently dimensional and adheres to a discipline that uses relational model with some important restrictions
  - The fundamental idea of dimensional modeling is that nearly every type of business data can be represented as a kind of cube of data
  - The model has also proved to provide high performance for queries and analysis

Dimensional Modeling Overview

- Components of Dimensional Model
  - Fact Table
    - The fact table contains facts or measurements of the business
  - Dimension Table
    - The dimension tables contain textual attributes that describe the facts

Dimensional Modeling Overview

Sample Report Translation

Sales Rep Performance Report		
Central Region		
	Jul-00	**Aug-00**
	(Dollars)	**(Dollars)**
Chicago District	879	878
Adams	345	456
Brown	564	565
Frederickson	657	768
Minneapolis District	890	789
Andersen Smith	909	978
Central Region Total	**4244**	**4434**

"Dimension"

Report,row and column heading

"Facts"

Numeric report values

Dimensional Modeling Overview

- Dimension Tables
  - Dimension Tables are the entry points into the data warehouse
  - Dimension tables are designed especially for selection and grouping under a common head
  - Determine contextual background for facts
  - Parameters for OLAP
  - Common Dimensions
    - Date
    - Product
    - Location/Region
    - Customers

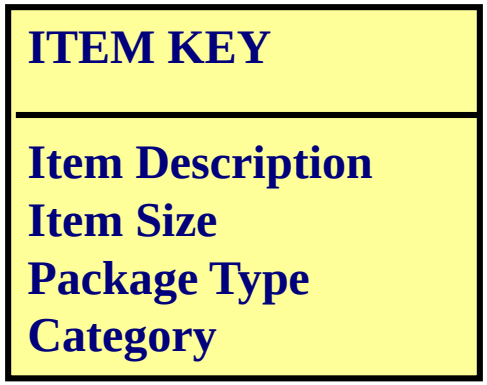
    The dimensional model to represent the information contained in the information package, the data structure must be represent with Metrics, business dimensions and attributes for each business dimension.

Dimensional Modeling Overview

- Dimension Tables
  - Dimension Table Characteristics
    - Serve as report labels and query constraints
      - “By” words
      - “Where” clauses
    - Provide Descriptive Information
      - Minimal codes
      - Embedded meaning as attributes
    - Represent hierarchical relationships

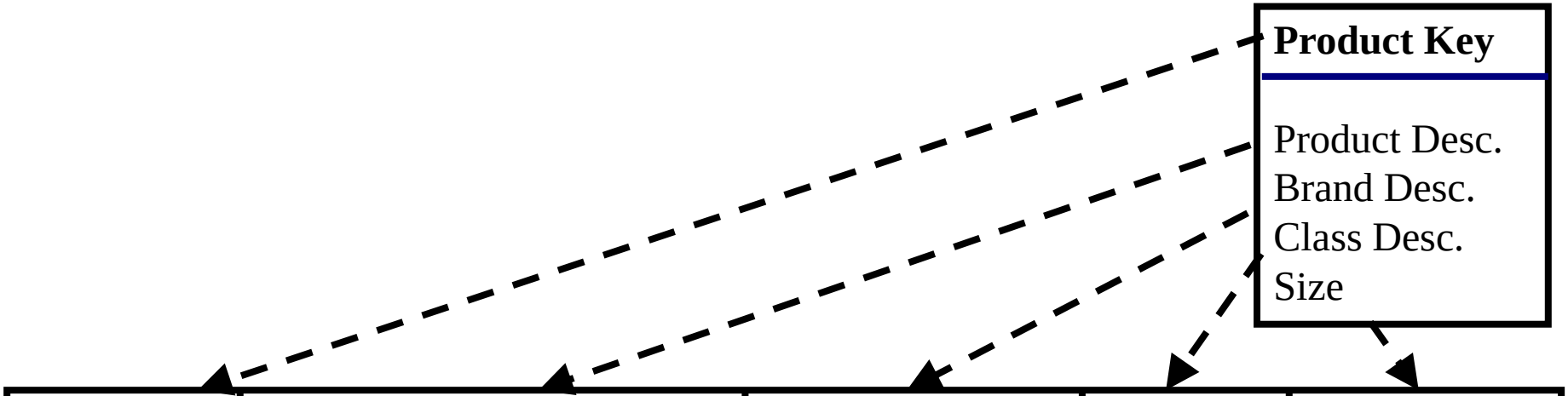
    Let see Product business dimension example,

    When we want to analyze the fact by products.



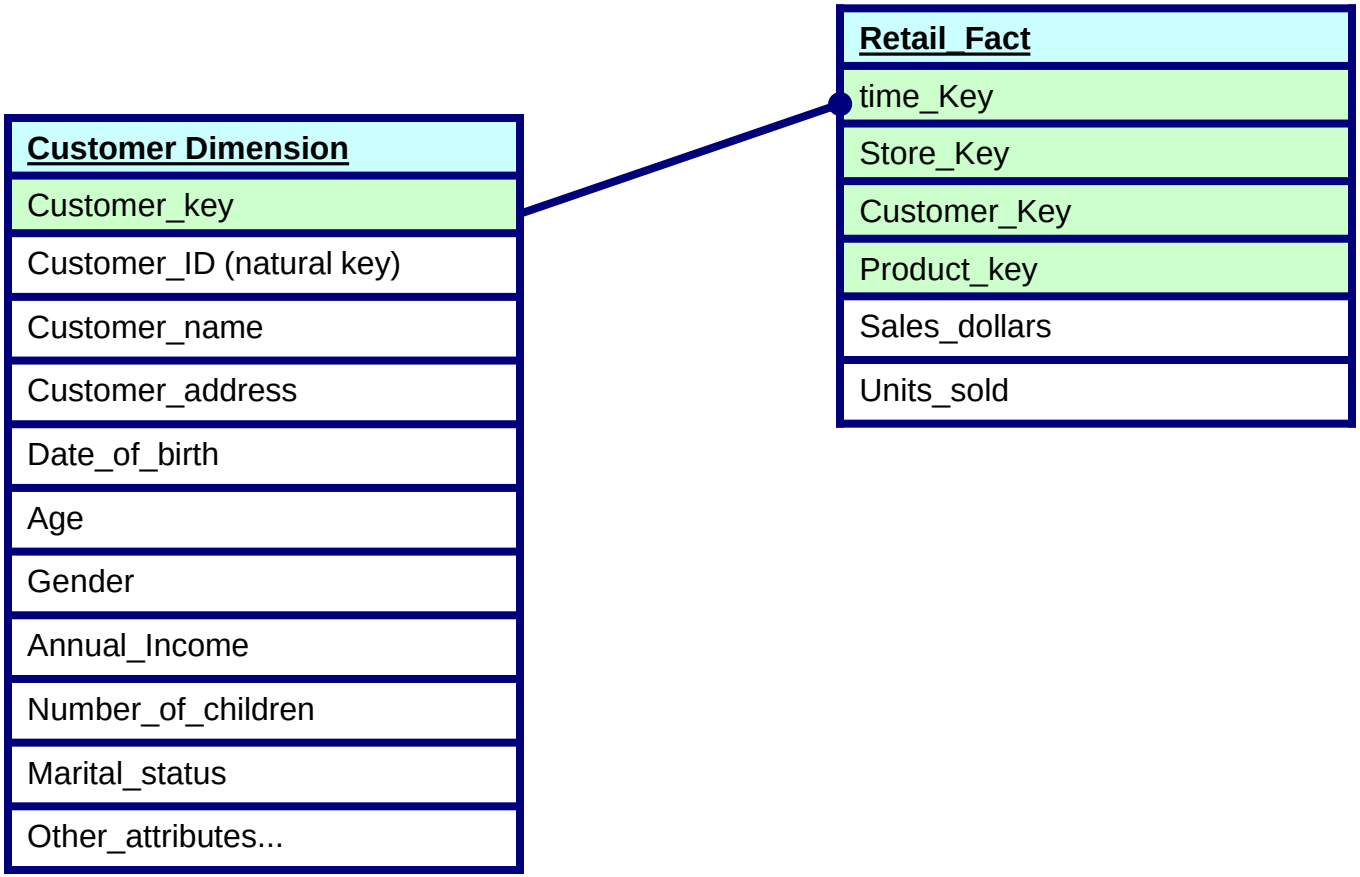
Dimensional Modeling Overview

Product Dimension Table with Sample Rows



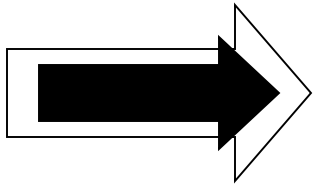
Product Key	Product Desc	Brand Desc	Class	Size
0001	CHEERIOS 10 OZ	CHEERIOS	Family	10 OZ
0002	CHEERIOS 24 OZ	CHEERIOS	Family	24 OZ
0003	LUCKY CHARMS 10 OZ	LUCKY CHARMS	Kids	10 OZ

Big Dimensions



Big Dimensions

Customer Dimension
Customer_key
Customer_ID (natural key)
Customer_name
Customer_address
Date_of_birth
Age
Gender
Annual_Income
Number_of_children
Marital_status
Other_attributes...



Customer Dimension
Customer_key
Customer_ID
Customer_name
Customer_address
Date_of_birth

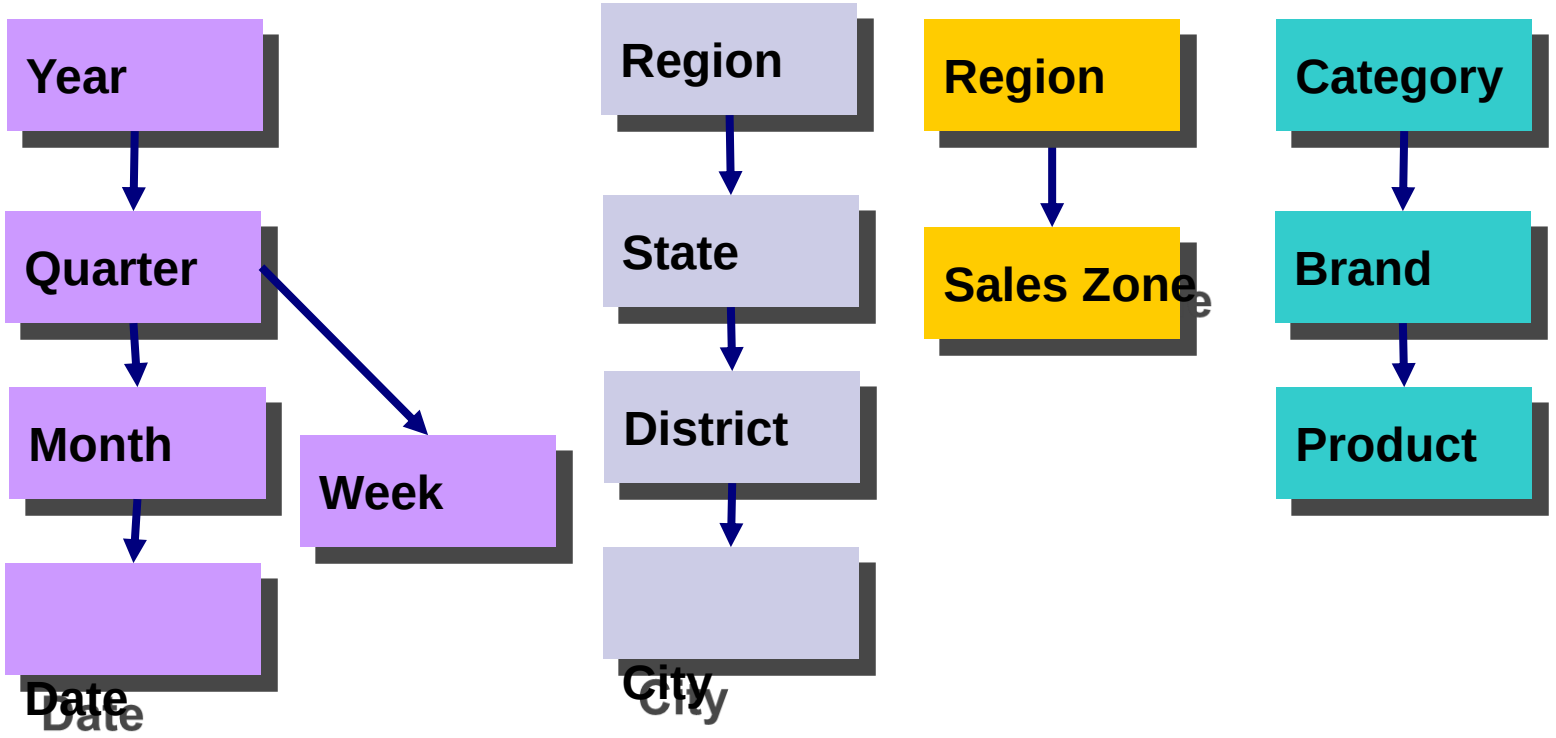
Customer Demographics Dimension
Customer_demographics_key
Age_Band
Gender
Income_Band
Number_of_children
Marital_status

Hierarchies in Dimensions

- Multiple Hierarchies
  - Dimension tables can represent multiple hierarchies roll-ups
  - For example ,Store Dimension could have
  - the following hierarchies
    - Physical Geography
      - Zip, City, County, State, Country
    - Sales Organization
      - District, Region, Zone
    - Distribution Roll-up
      - Distribution Center, Distribution Center R

Store Dimension
Store_key
Store_description
Store_type
Zip
City
State
Sales_region
Sales_zone

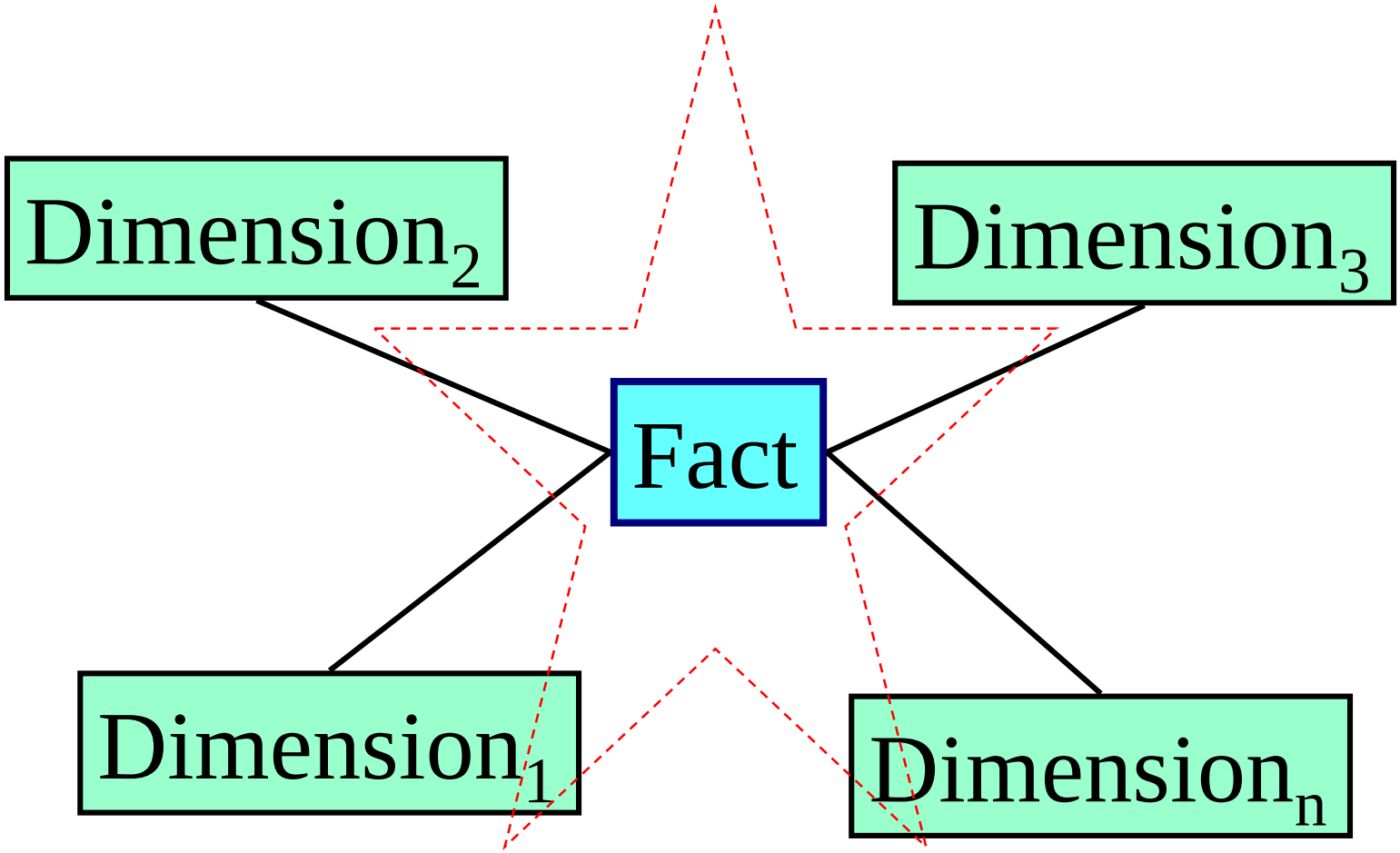
Hierarchies in Dimensions



Dimensional Modeling Overview

- So far we have formed fact table and dimension tables.
- How should these tables be arranged in the dimensional model?
- What are the relationships and how should we mark the relationships in the model?
- The dimensional model should primarily facilitate queries and analyses. What would be the types of queries and analyses?
- Before combining these tables in dimensional model. What are the requirements?
  - The model should provide the best data access.
  - The whole model must be query-centric.
  - It must be optimized for queries and analysis.
  - The model must show that the dimension tables interact with the fact table.
  - It should structured in such a way that every dimension can interact equally with the fact table.
  - The model should allow drilling down or rolling up along dimension hierarchies.
  - With this rqts., each of the dimension tables are directly relates to fact table in the middle.
- Such an arrangement in the dimensional model looks like a star formation, with the fact table at the core of a star and the dimension tables along the spikes of the star.
- The dimensional model is therefore called a STAR schema.

Star like Model



Star model…..

- It consists of sales fact table in the middle of schema diagram. It have 3 dimension tables of Date, Product and Store.
- The user will analyze the sales using dollar sold, unit sold and dollar cost.
- From the STAR schema structure intuitively answers the questions for a given amount of dollars, what was the product sold? Who was the customer? Which store sold the product? When was the order placed?
- Constraints and filters of queries are easily understood by looking at the star.
- A common type of analysis is the drilling down the summary numbers to get at the details at the lower levels by filtering queries.

Entity-Relationship vs. Dimensional Models – Contrasting

- One table per entity
- Minimize data redundancy
- Optimize update
- The Transaction Model
- highly normalized model

- One fact table for data organization
- Maximize understandability
- Optimized for retrieval
- The data warehousing model
- Highly aggregate model

Design Issues in Comparison

Relational and Multidimensional Models

- Denormalized and indexed relational models more flexible
- Multidimensional models simpler to use and more efficient

E-R vs. Dimensional Models – as Complimentary

- E-R Model & Dim Model: goes in-hand
- Both work on the base concept of RDBMS with structural variations
- Both follow the theory of Referential Integrity constraint, but in differnet manner
- Dimensional Model is an extension of E-R Model, it will act like input to Dim Model
- No existence of Dimensional Model out with proper support of Data Model (E-R)

Relation btw ER and DM

- **The key to understand the relationship between DM and ER is that a single ER diagram breaks down into multiple DM diagrams**
- **For suppose large ER diagram as representing every possible business process in the enterprise.**

  - Then the **first** step in converting an ER diagram to a set of DM diagrams

  - And into discrete business processes and to model each one separately

Relation btw ER and DM

- The **second** step is to select those many-to-many relationships in the ER model containing numeric and additive non-key facts and to designate them as fact tables.
- The **third** step is to de-normalize all of the remaining tables into flat tables with single-part keys that connect directly to the fact tables. - These tables become the dimension tables.
- **If the dimension table connects to more than one fact table, we represent this same dimension table in both schemas, and we refer to the dimension tables as "conformed" between the two dimensional models.**

Can DM Models Represent E-R Models?

- **Yes**, in case if It depends on the relationships in the conceptual model formalized by the logical data model.
- E-R model can be defined whether or not many-many relationships exist. But without them it would have no fact tables.
- whether an **E-R data warehouse model can always be represented as a series of dimensional models**. depends on whether the underlying conceptual model of a data warehouse must always contain many-to-many relationships. - If it follows that an E-R data warehouse can be expressed as a star schema.
- The **fact tables are resolved many-many relations** among fundamental entities, it follows that in a correct E-R model
- The fact tables are a necessary consequence of grain attributes and of standard E-R modeling rules requiring, conceptual correctness and syntactic completeness.

Fact Tables

- Contains two or more foreign keys
- Tend to have huge numbers of records
- Useful facts tend to be numeric and additive

Dimension Tables

- Contain text and descriptive information
- 1 in a 1-M relationship
- Generally the source of interesting constraints
- Typically contain the attributes for the SQL answer set.

Strengths of the Dimensional Model

- Predictable, standard framework
- Respond well to changes in user reporting needs
- Relatively easy to add data without reloading tables
- Standard design approaches have been developed
- There exist a number of products supporting the dimensional model

Business Model

As always in life, there are some disadvantages to 3NF:

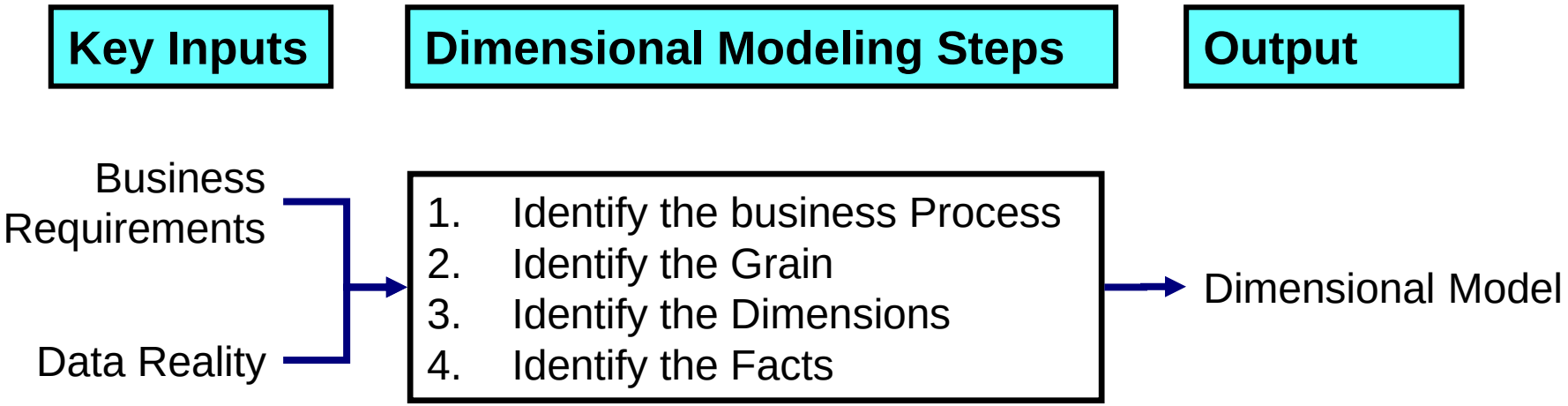
- Performance can be truly awful. Most of the work that is performed on denormalizing a data model is an attempt to reach performance objectives.
- The structure can be overwhelmingly complex. We may wind up creating many small relations which the user might think of as a single relation or group of data.

The Business Model

Identify the data structure, attributes and constraints for the client's data warehousing environment.

- Stable
- Optimized for update
- Flexible

Dimensional Modeling Steps



The 4 Step Design Process

- Choose the Data Mart
- Declare the Grain
- Choose the Dimensions
- Choose the Facts

Dimensional Modeling Steps

- Identify the Business Process
  - A major operational process that is supported by some kind of legacy system(s) from which data can be collected for the purpose of the data warehouse
  - Example: orders, invoices, shipments, inventory, sales
- Identify the Grain
  - The fundamental lowest level of data represented in a fact table for the business process
  - Example: individual transactions, individual daily snapshots
- Identify the Dimensions
  - Choose the dimensions that will apply to each fact table record
- Identify the Facts
  - Choose the measured facts that will populate each fact table record

Building a Data Warehouse from a Normalized Database

The steps

- Develop a normalized entity-relationship business model of the data warehouse.
- Translate this into a dimensional model. This step reflects the information and analytical characteristics of the data warehouse.
- Translate this into the physical model. This reflects the changes necessary to reach the stated performance objectives.

Structural Dimensions

- The first step is the development of the structural dimensions. This step corresponds very closely to what we normally do in a relational database.
- The star architecture that we will develop here depends upon taking the central intersection entities as the fact tables and building the foreign key => primary key relations as dimensions.

Steps in dimensional modeling

- Select an associative entity for a fact table
- Determine granularity
- Replace operational keys with surrogate keys
- Promote the keys from all hierarchies to the fact table
- Add date dimension
- Split all compound attributes
- Add necessary categorical dimensions
- Fact (varies with time) / Attribute (constant)

Converting an E-R Diagram

- Determine the purpose of the mart
- Identify an association table as the central fact table
- Determine facts to be included
- Replace all keys with surrogate keys
- Promote foreign keys in related tables to the fact table
- Add time dimension
- Refine the dimension tables

Choosing the Mart

- A set of related fact and dimension tables
- Single source or multiple source
- Conformed dimensions
- Typically have a fact table for each process

Grain (unit of analysis)

The grain determines what each fact record represents: the level of detail.

- For example
  - Individual transactions
  - Snapshots (points in time)
  - Line items on a document
- Generally better to focus on the smallest grain

Dimensions

A table (or hierarchy of tables) connected with the fact table with keys and foreign keys

- Preferably single valued for each fact record (1:m)
- Connected with surrogate (generated) keys, not operational keys
- Dimension tables contain text or numeric attributes

Fact Tables

Represent a process or reporting environment that is of value to the organization

- It is important to determine the identity of the fact table and specify exactly what it represents.
- Typically correspond to an associative entity in the E-R model

Facts

Measurements associated with fact table records at fact table granularity

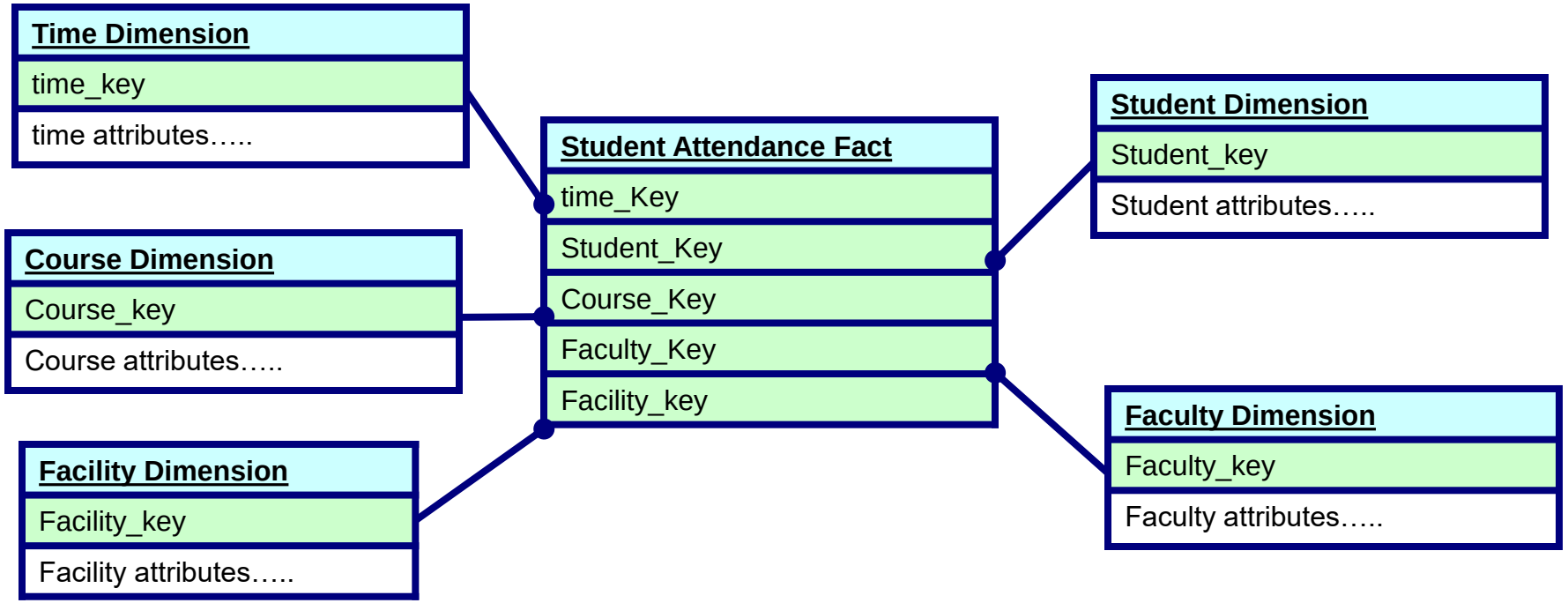
- Normally numeric and additive
- Non-key attributes in the fact table

  Attributes in dimension tables are constants. Facts vary with the granularity of the fact table

Additive Measures

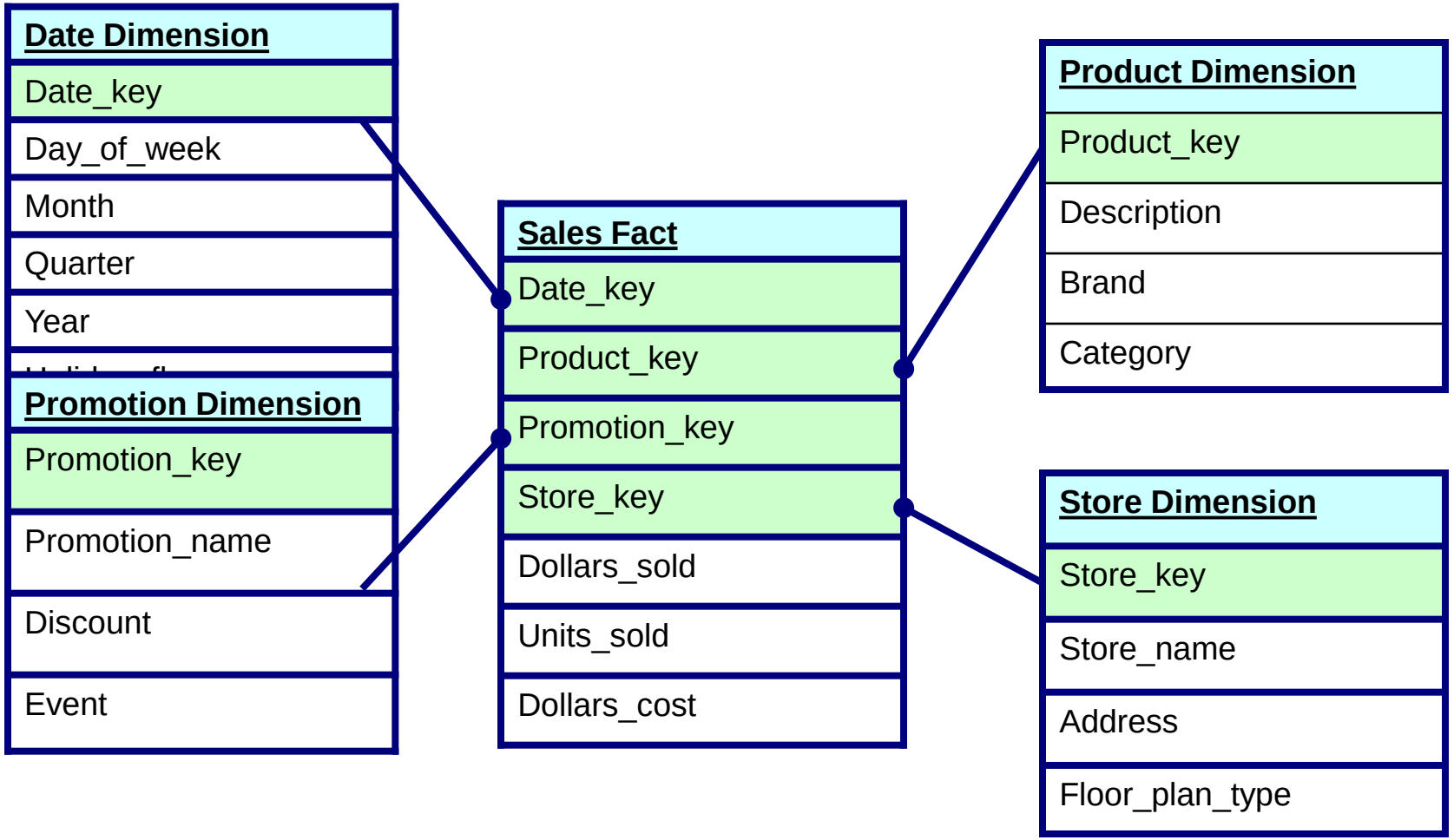
- The ability of measures to be added across all dimensions of the fact table.
- Measures could be fully additive, semi additive or non additive
  - Fully Additive - The values of the attributes summed up by simple addition, Aggregation is a fully additive measures is done by simple addition. Sales Quantity, Revenue
  - Semi Additive - Account Balance, Inventory, number of customers (Measure of Intensity, head counts)
  - Non-Additive - Profit margin (Ratios and Percentages) i.e., Ratio or Percentages should not be added 1:3, 30%, etc.

Factless Fact tables to track attendance of students

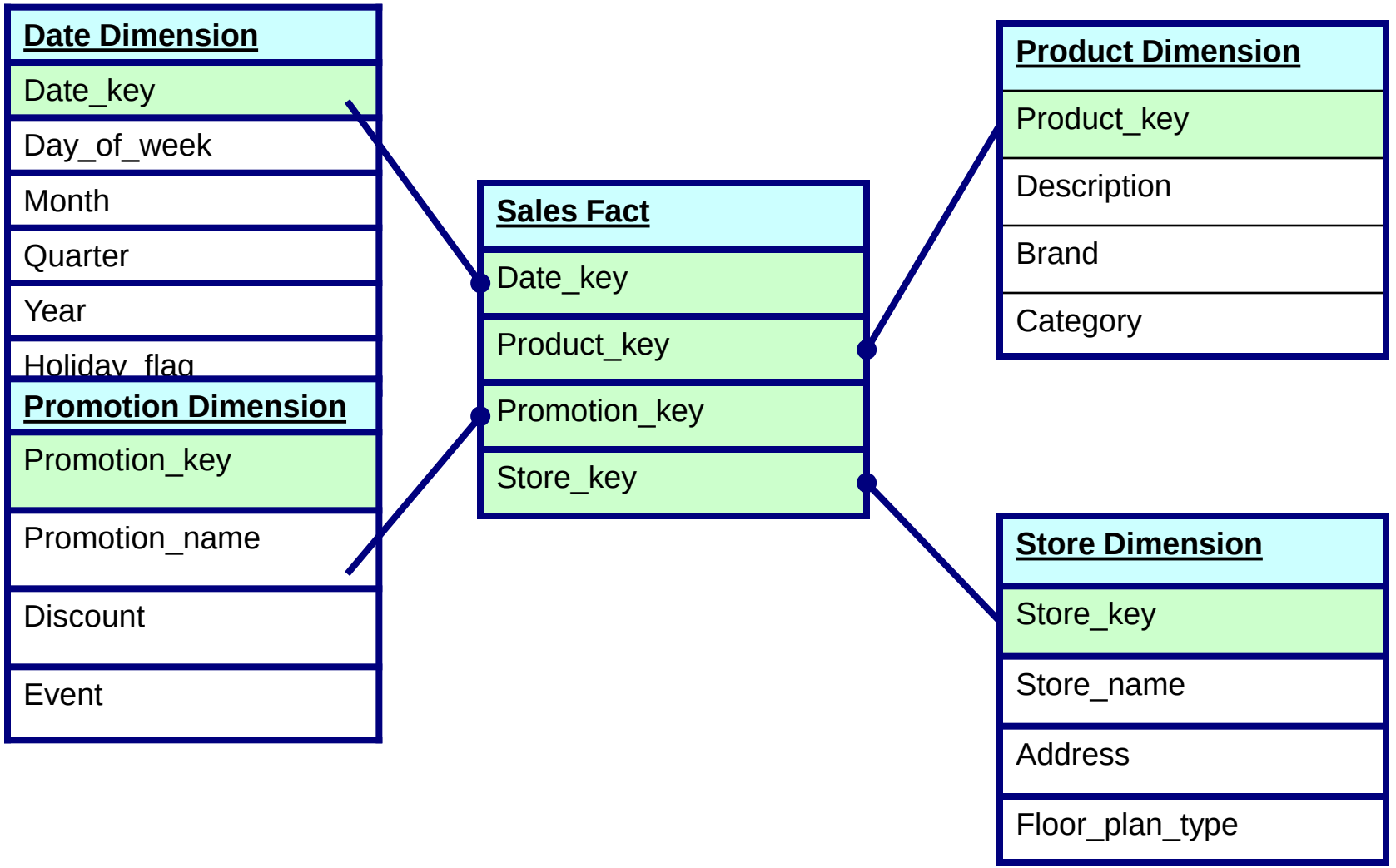


Factless Fact tables – Coverage tables

Sales Fact (revisited)



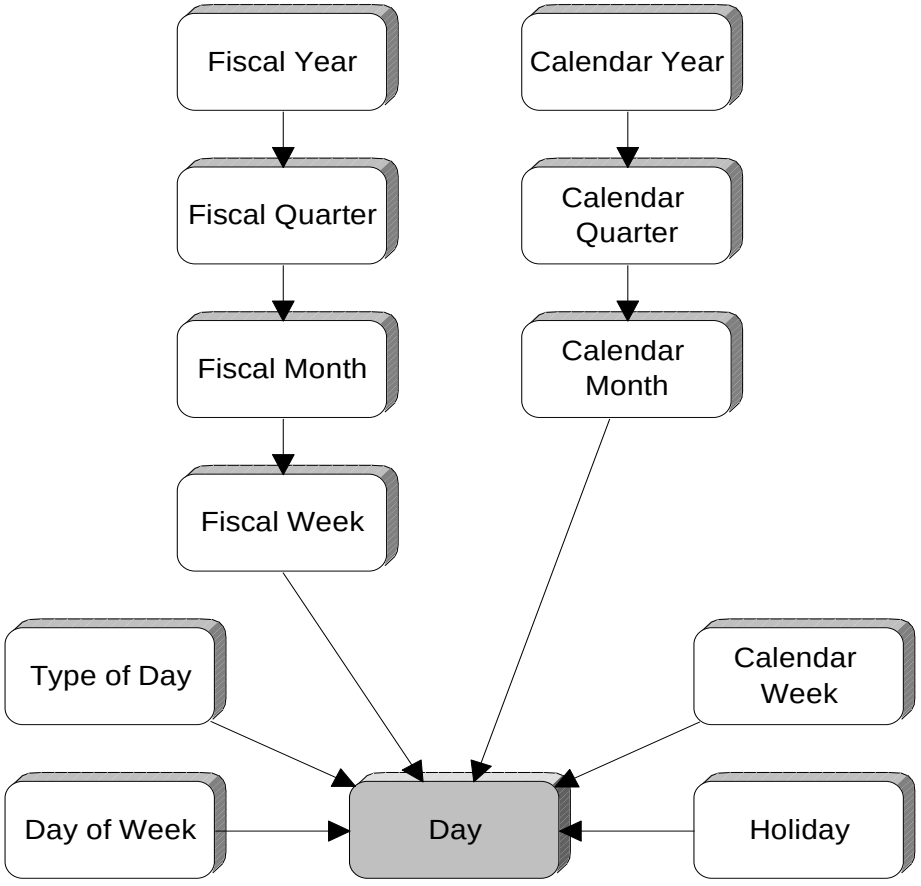
Factless Fact tables – Coverage tables



Snowflaking & Hierarchies

- Efficiency vs Space
- Understandability
- M:N relationships

Example: Date Dimensions

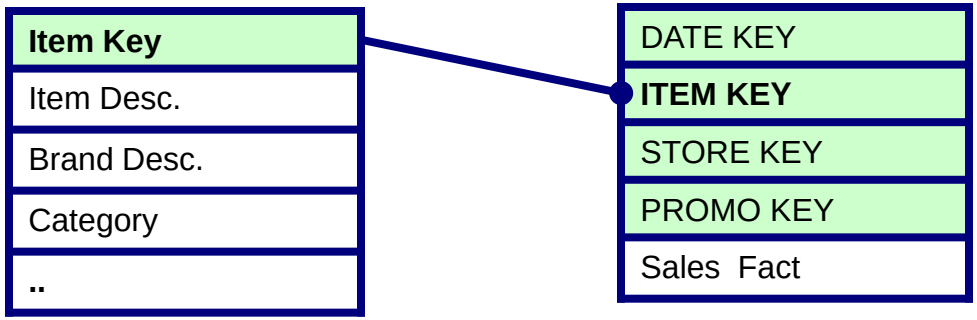


Conformed Dimension

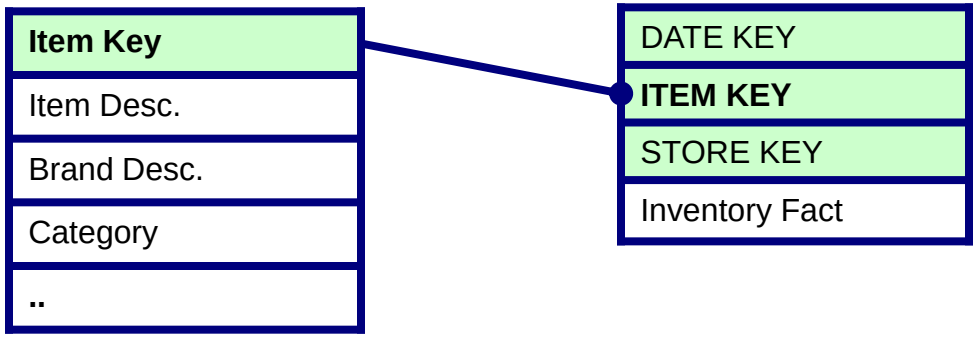
Shared Dimensions Must Conform

- **Option 1:** Identical dimensions with the same keys, labels, definitions and values

Sales Schema



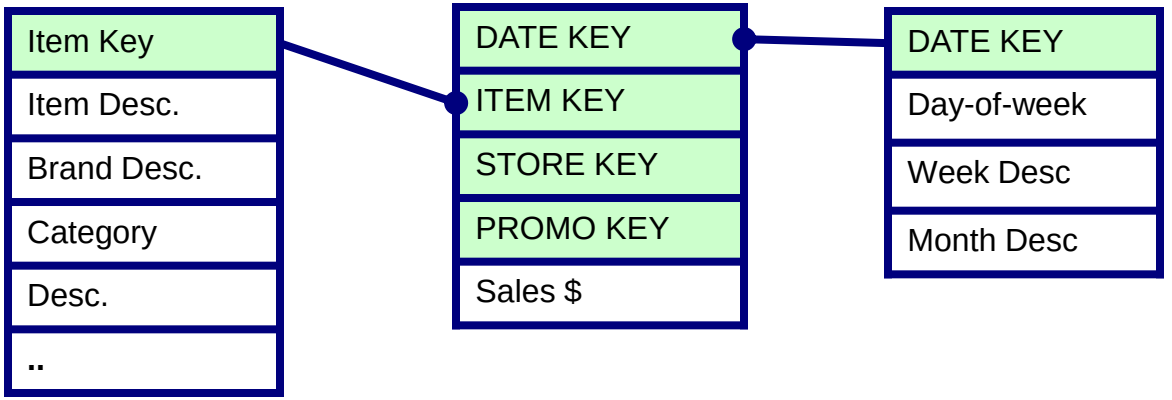
Inventory Schema



Conformed Dimension

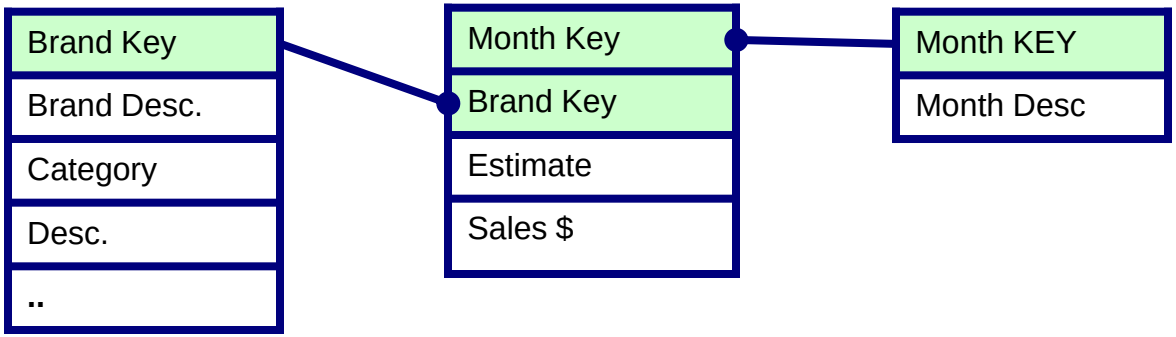
- **Option 2:** "Subset" of base dimension

Sales Schema



Item key	Item Desc	Brand Desc	Category Desc
0001	Cheerios 10oz	Cheerios	Cereal

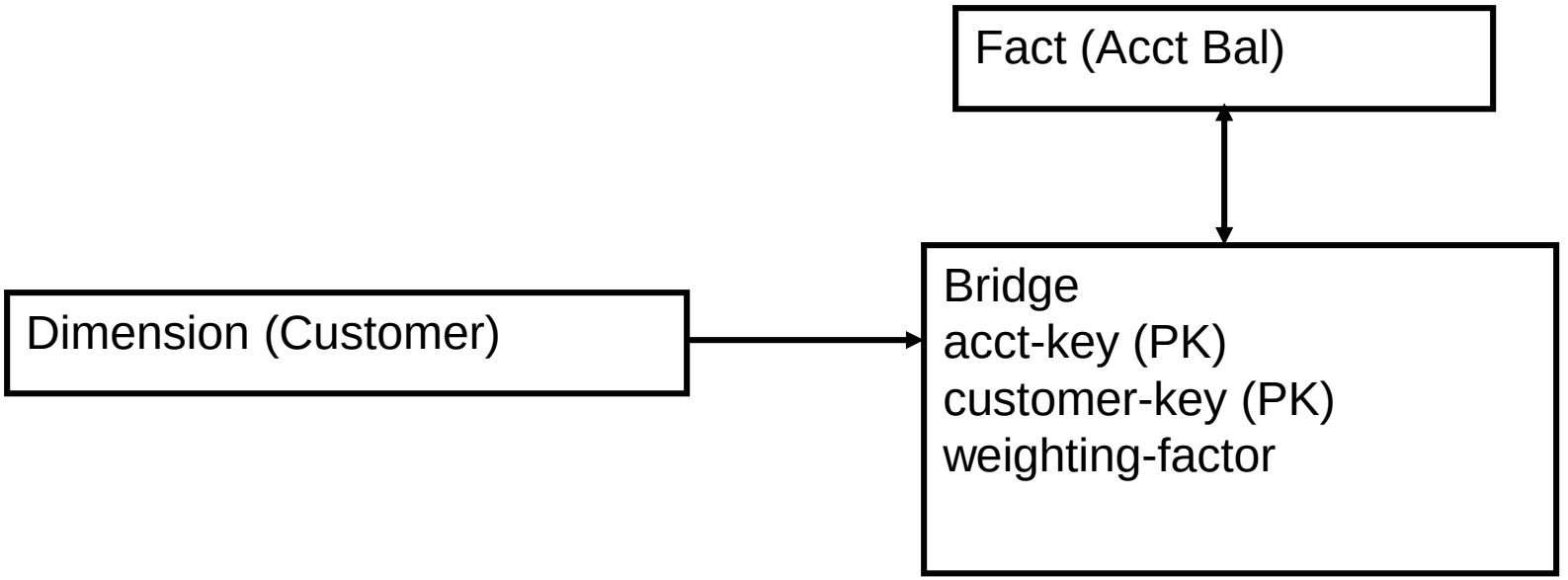
Forecast Schema



Brand key	Brand Desc	Category Desc
1001	Cheerios	Cereal

Many to many Overcome

- Use a Bridge Table
- Add a weighting factor to correct fact addition



Slowly Changing Dimensions

(Addresses, Managers, etc.)

- Type 1: Store only the current value
- Type 2: Create a dimension record for each value (with or without date stamps)
- Type 3: Create an attribute in the dimension record for previous value

Slowly Changing Dimensions

- Dimension attributes evolve over time
  - For example, customers change their names, move, have children, adjust their Incomes
- For every dimension attribute, need to identify “Changes” strategy
  - May use combination of strategies within same dimension table

Slowly Changing Dimensions

Type 1 : Overwrite the changed attributes

Original record

Item Key	Item Desc	Dept
12345	Sim City 3000	Educational S/W

Updated record

Item Key	Item Desc	Dept
12345	Sim City 3000	Strategy S/W

Slowly Changing Dimensions

Type 2 : Add a New Dimension Record

Original record

Item Key	Item Desc	Dept
12345	Sim City 3000	Educational S/W

Additional record

Item Key	Item Desc	Dept
12345	Sim City 3000	Strategy S/W

Slowly Changing Dimensions

Type 3 : Add a "Prior" Attribute

Original record

Item Key	Item Desc	Dept
12345	Sim City 3000	Educational S/W

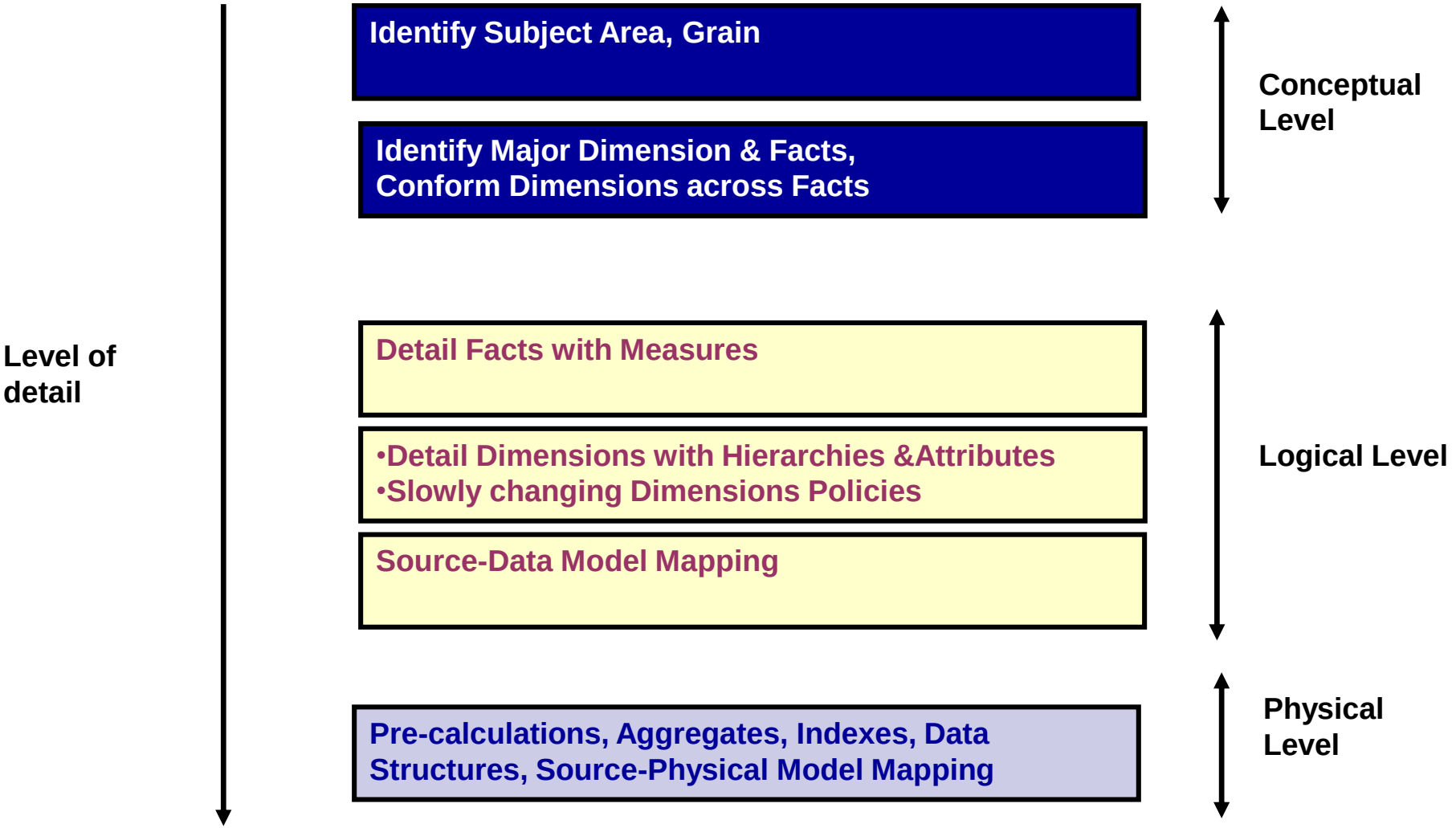
Updated record

Item Key	Item Desc	Dept	Prior Dept
12345	Sim City 3000	Strategy S/W	Educational S/W

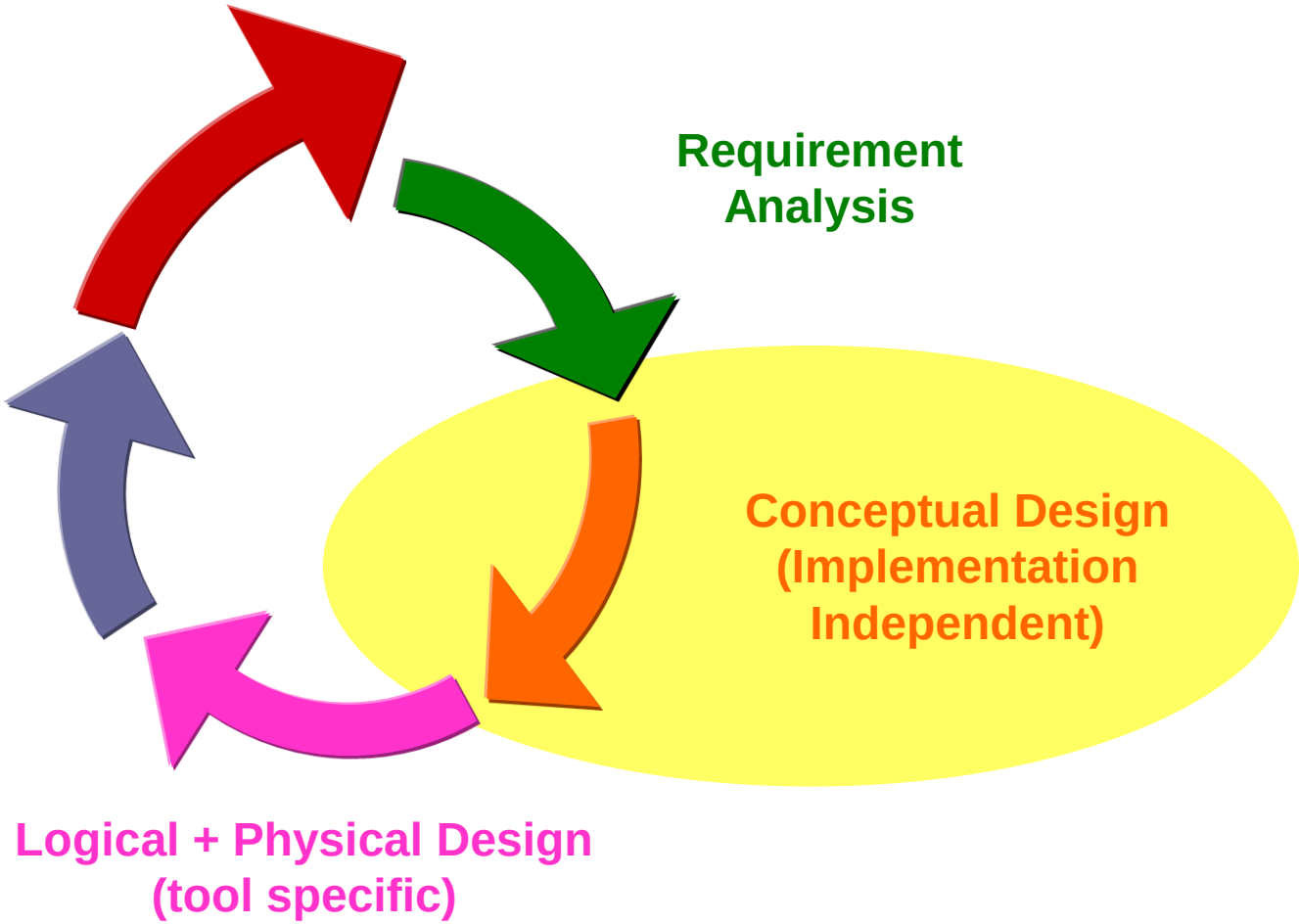
Slowly Changing Dimensions

- Data Warehouse Keys ie., STAR schema keys
  - All tables (facts and dimensions) should use Data Warehouse generated **surrogate keys**
  - It is possible that the customer no., of discontinued customers are reassigned to new customers. We will have a problem because the same customer no,. Could relate to the data for the newer customer and also to the data of the retried customer. Therefore do not use such keys as a primary keys for dimensional tables.
  - The surrogate keys are simply system generated sequence numbers.
  - Each row in a dimension table is identified by a unique value of an attribute designated as the **primary key** of the dimension.
  - Each dimension table is in 1:M relationship with central fact table. So the primary key of each dimension table must be a **foreign key** in the fact table.

Dimensional Modeling Framework/Flow Example



Conceptual Design



The Modeling Process

- Which business process is being modeled?
- What is the subject of analysis (fact) and what is being measured?
- On what granularity level is active analysis being done?
- Which properties (dimensions) determine the measures?
- Which different levels of aggregation are meaningful?
- What additional information is needed for the different levels?
- What is the variability and the cardinality of the dimensions?

Facts

- Fact = Subject of Analysis — Sales
- Measures = Attributes describing facts

  Quantity, Price
- Derived Measures — Profit
- Additivity of Measures
  - globally additiv — Quantity
  - additiv for some dimensions — Items in stock

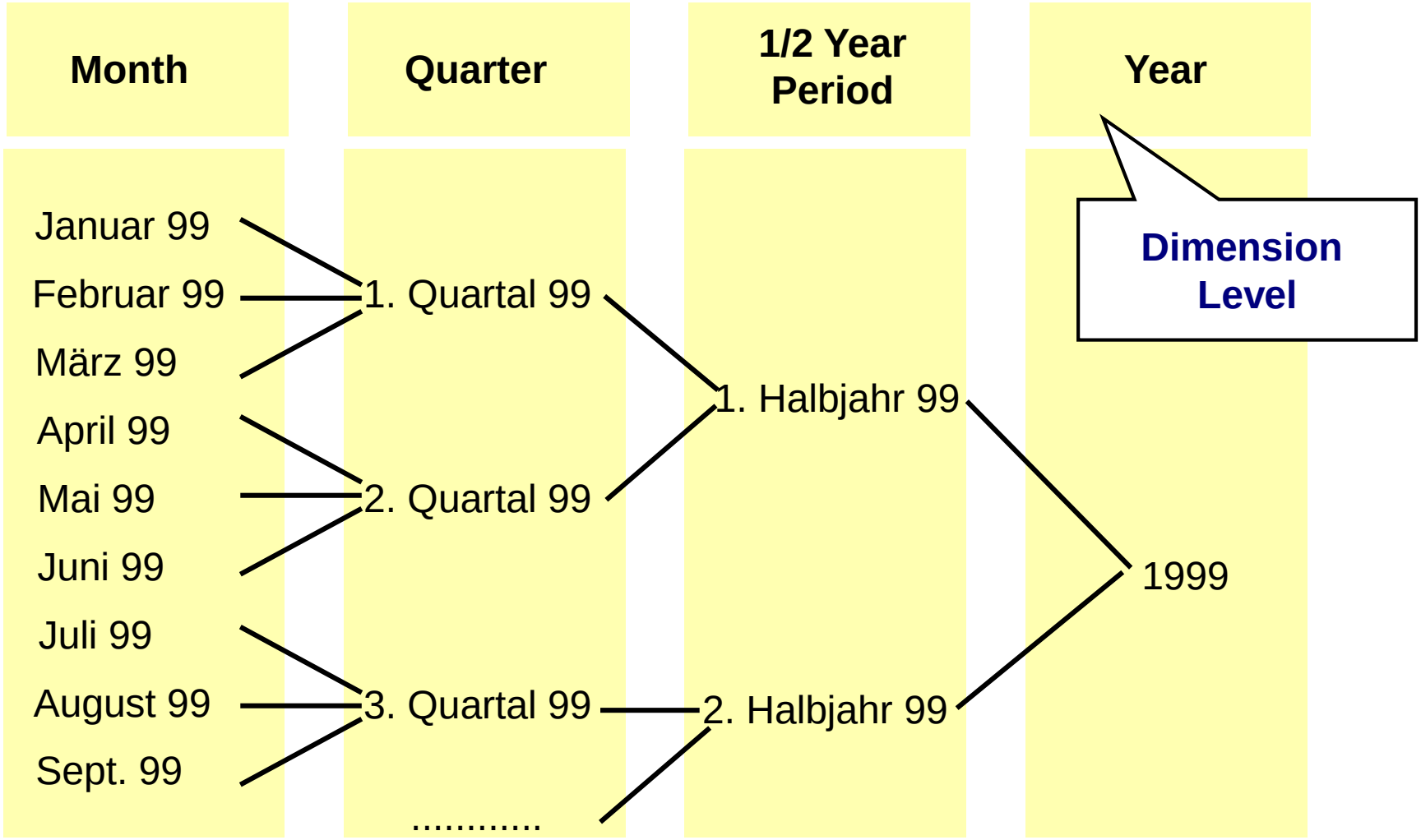
    additiv resp. to plants/

    not additiv w.r.t. time
  - not additiv at all — profit margin

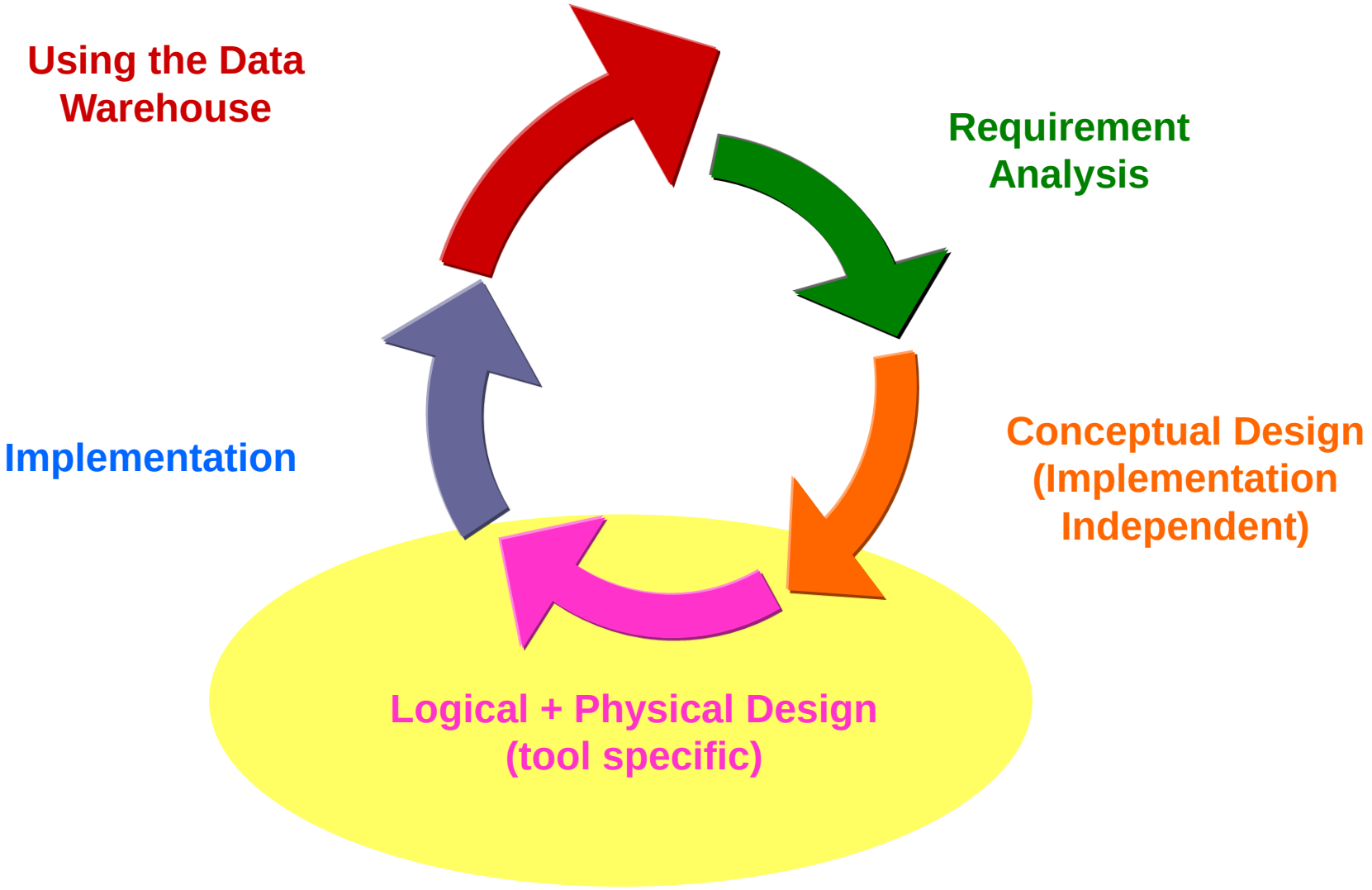
Dimensions

- Dimensions = static structure of business information
- Used for navigating the data space
- Choosing the necessary granularity
- Dimension Members = Instances of a dimension
  - e.g. 8.12.1997 and Juli 1997 are members of dimension “time”
- Structuring Dimension
  - using different dimension levels (hierarchies)
  - using descriptive attributes

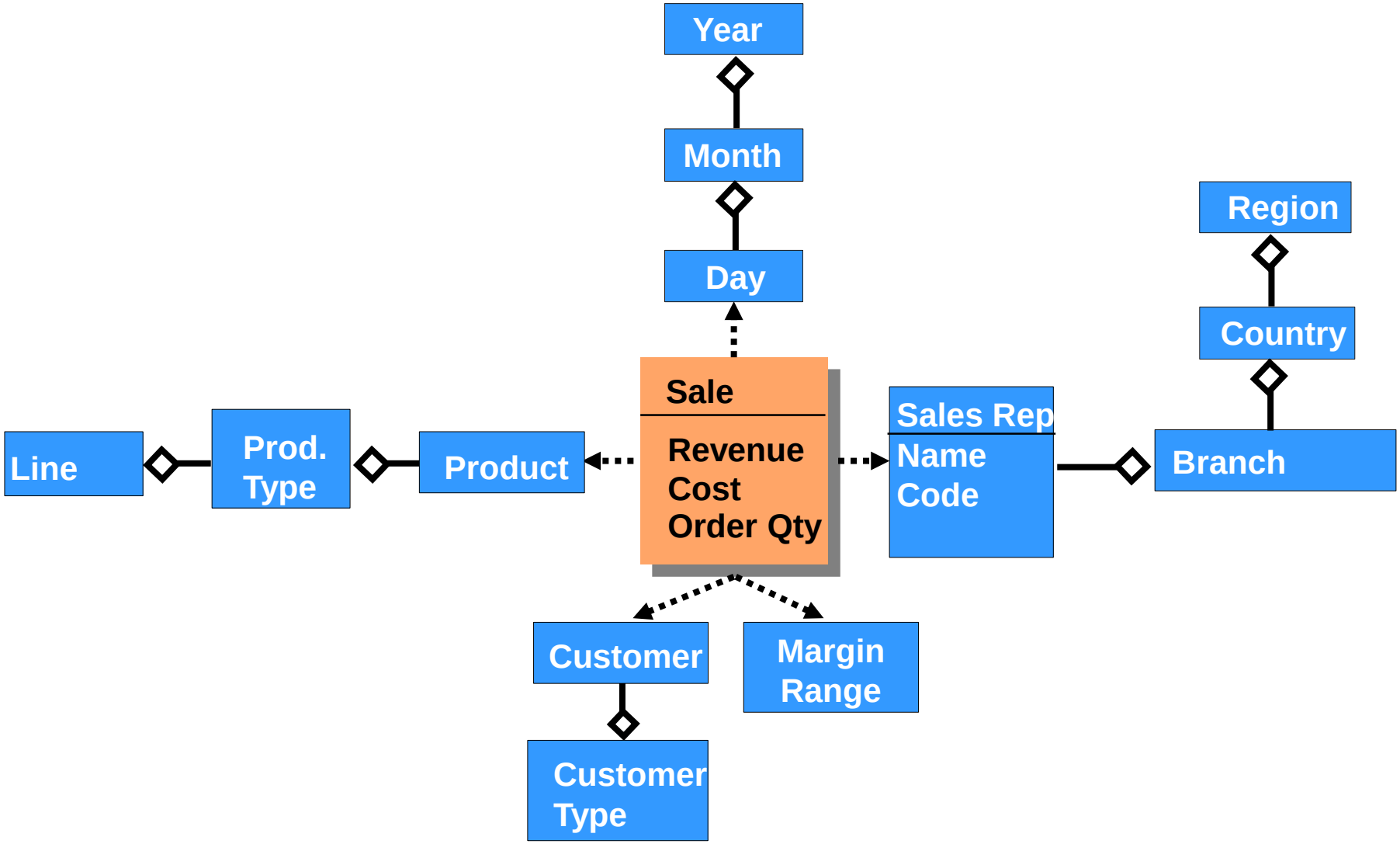
Simple Hierarchies



Logical + Physical Design

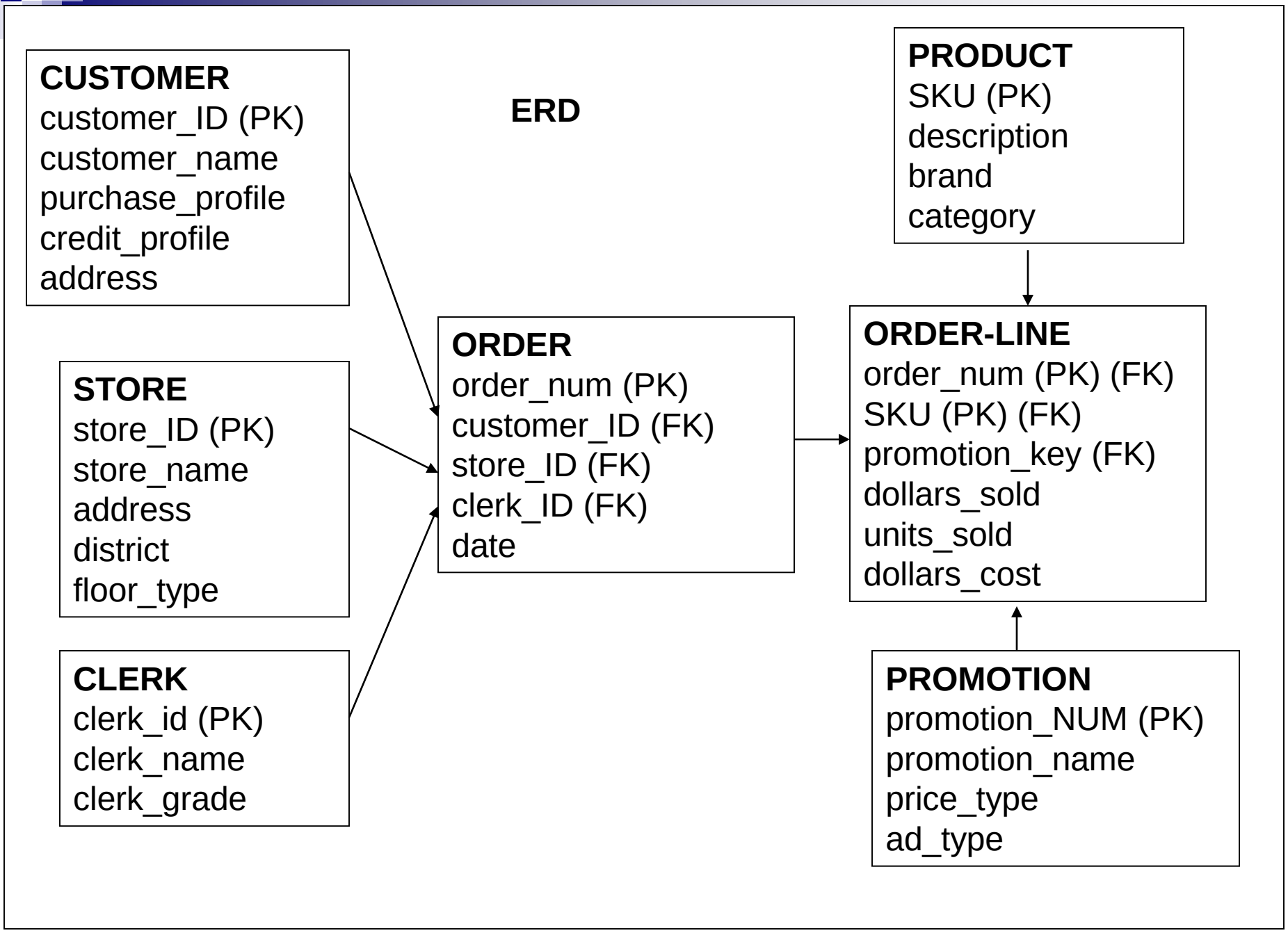


Example Data Model



Dimensional Modeling Tips

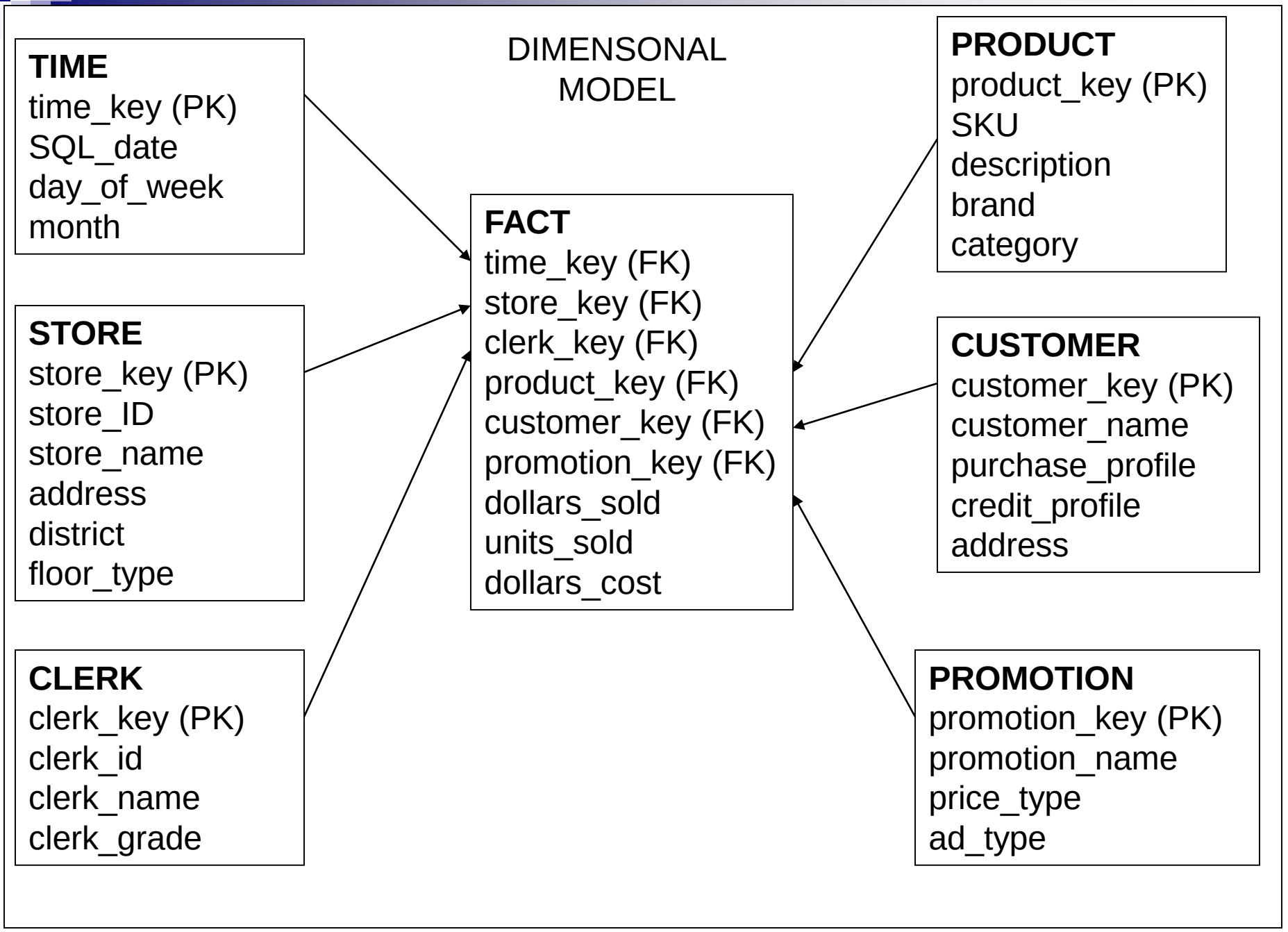
- Carefully choose the labels to identify data marts, dimension, attributes and facts
- An attribute can live in one and only one dimension, whereas a fact can be repeated in multiple fact tables
- If a single dimension appears to reside in more than one places, several roles are probably being played. Name the roles uniquely and treat them as separate dimensions
- A single field in the underlying source data can have one or more logical columns associated with it
  - E.g., A product attribute field may translate to product code, product short description, and product long description
- Every fact should have a default aggregation rule (sum, min, max, latest, semi-additive, special algorithm, and not aggregatable)
  - This will serve as a requirements list for query and report writers tools evaluations


ERD
CUSTOMER
customer_ID (PK)
customer_name
purchase_profile
credit_profile
address
PRODUCT
SKU (PK)
description
brand
category
STORE
store_ID (PK)
store_name
address
district
floor_type
ORDER
order_num (PK)
customer_ID (FK)
store_ID (FK)
clerk_ID (FK)
date
ORDER-LINE
order_num (PK) (FK)
SKU (PK) (FK)
promotion_key (FK)
dollars_sold
units_sold
dollars_cost
CLERK
clerk_id (PK)
clerk_name
clerk_grade
PROMOTION
promotion_NUM (PK)
promotion_name
price_type
ad_type

Apply Normalize Method -Example

- Applying Normalize on building E-R Model in Identifying Entities and Attributes
- Demonstrate with example, the flow of 1st to 3rd Normal forms in E-R Model
- Check with the Case Study design Flow

-- Please the Day 4 under Dwh TOC


DIMENSONAL MODEL
TIME
time_key (PK)
SQL_date
day_of_week
month
STORE
store_key (PK)
store_ID
store_name
address
district
floor_type
CLERK
clerk_key (PK)
clerk_id
clerk_name
clerk_grade
FACT
time_key (FK)
store_key (FK)
clerk_key (FK)
product_key (FK)
customer_key (FK)
promotion_key (FK)
dollars_sold
units_sold
dollars_cost
PRODUCT
product_key (PK)
SKU
description
brand
category
CUSTOMER
customer_key (PK)
customer_name
purchase_profile
credit_profile
address
PROMOTION
promotion_key (PK)
promotion_name
price_type
ad_type

Apply De-normalize Method -Example

- Presenting De-Normalize Method on Dimensional Model Design process
- Demonstrate with example on Identifying Fact & Dimension & Hierarchy as a part of Dim Model build from Existing E-R Model
- Check with the Case Study design Flow

-- Please check the Day 4 under DWH TOC