What's Ahead?

Is Dimensional Modeling Still a Thing?

What is Dimensional Modeling?

How It Differs from Relational Modeling

Dimensions

Facts

The Star Schema

The Snowflake Schema

Why a Data Warehouse?

Consistency/Quality (Regulatory) – Core Business Data

Governance

Verifiability

Supportability

Usability

History

When is it Not Appropriate?

Unstructured Data/Not Core Business

Short Term Goals

Extremely Quick Extensibility

Silo Based

Transactional Database Design (OLTP)

OLTP Performance is about Inserting and Updating Quickly

Locking Must Be Minimized

Very Small Sets of Data Is Retrieved in a Query

Data Consistency is Critical

Laws of Normalization

Focus is on the customer(s) entering data.

Reporting Database Design (Data Warehouse)

Copy of OLTP Data

Performance is about retrieving the data quickly.

Locking is not an issue.

Large sets of data are retrieved in a query.

Insert and Update speed is not important.

Focus is on the End User Running Queries

What is Dimensional Modeling?

"Dimensional Modeling is a design technique for databases intended to support end-user queries in a data warehouse"

Ralph Kimball

What is Dimensional Modeling?

Data maintenance performance is secondary.

Data is denormalized as needed to support reporting.

The resulting model reflects the kinds of questions the business wants to ask rather than the functions of the underlying operational system.

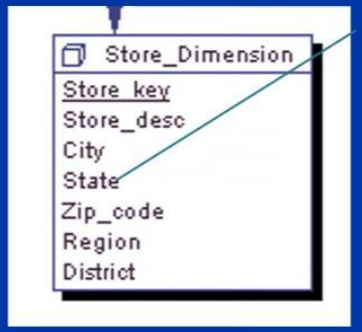
Descriptive data like customer name is separated from the quantity data such as order amount.

What's a Dimension?

What is a Dimension?

- Dimensions describe business events like the sale of a product.
- They are what users would want to sort, group and filter on like dates, customer number, store number, etc.

An example of a dimension…



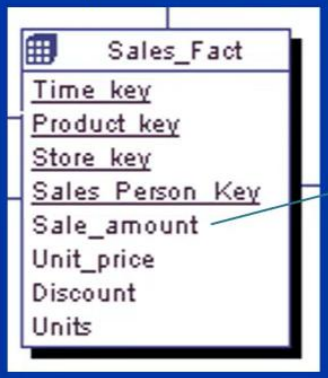
A user might want to group by store state.

What's a Fact?

What is a Fact?

- A fact, also called a measure, is a measureable metric which is described by the dimensions such as the sale amount or order quantity.
- There are usually many more dimensions than facts.

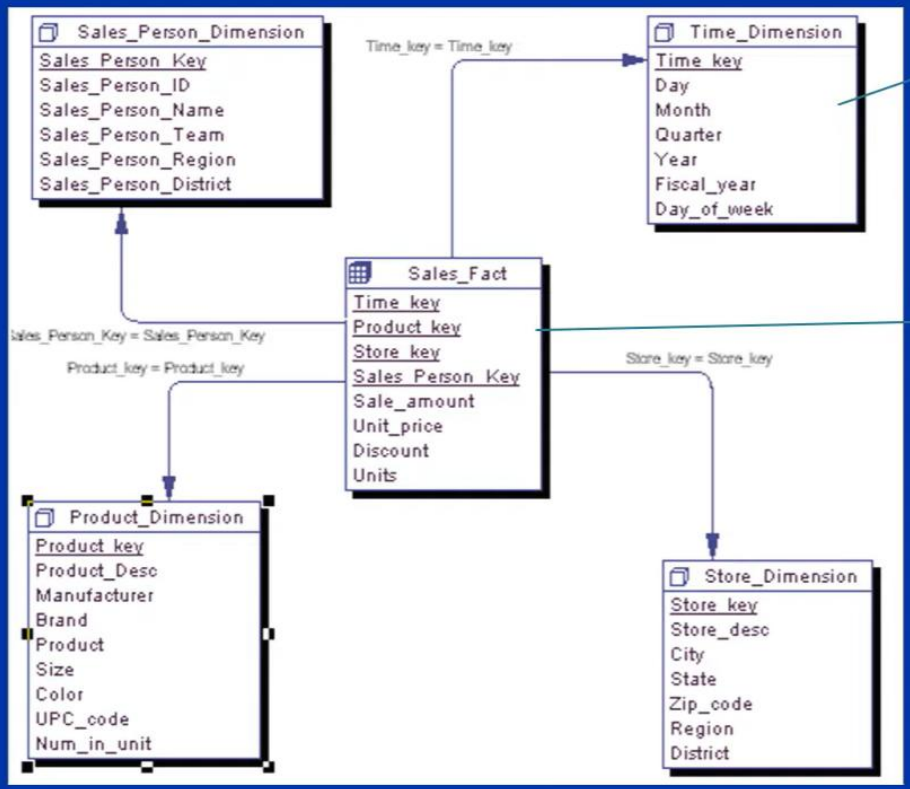
An example of a fact table...



Sales amount is a fact. We would likely want to summarize it.

The Star Schema

The Star Schema

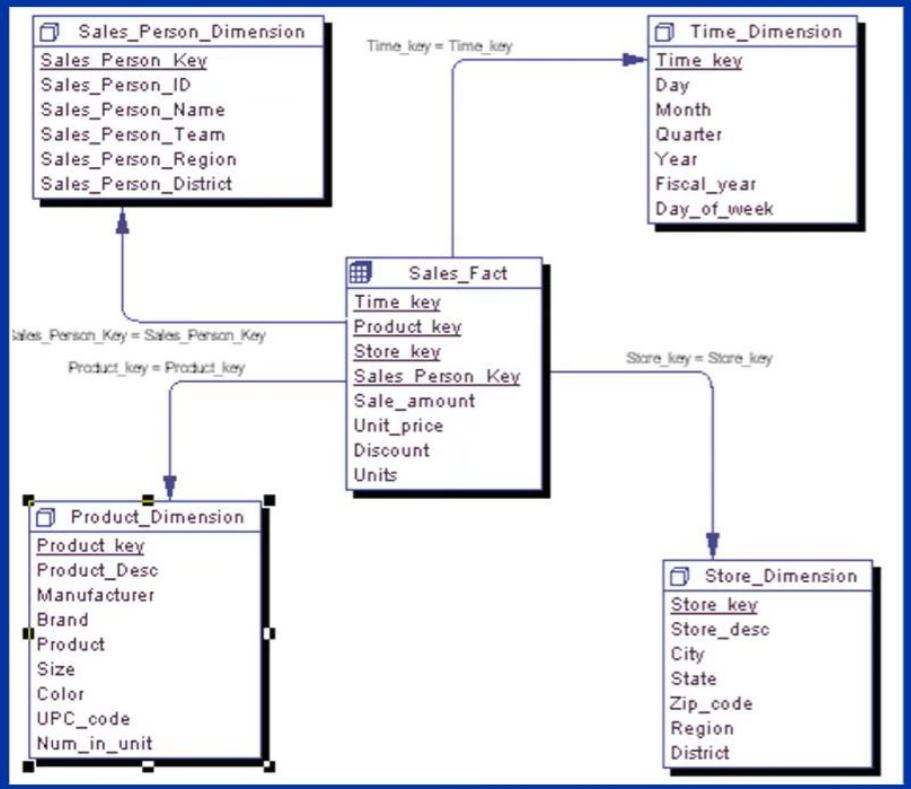


An example of a fact table…

A dimension can contain many dimension attributes.

A fact table has facts (or measures) and a key to each related dimension.

Star Schema – Key Points



- Dimensions relate directly to the fact table only.
- The dimensions are denormalized, i.e. Sales_Person_Region does not have a related region lookup table as an OLTP design would likely have.
- Usually the dimension keys are NOT keys from the source systems, rather they are generated by the data warehouse load process and they are called surrogate keys.
- The dimension attributes you define determine the granularity called the grain of the facts, i.e. how detailed are the measures.
- Warning! This is not a relational design so careful if you are an OLTP developer.

From Relational to Dimensional

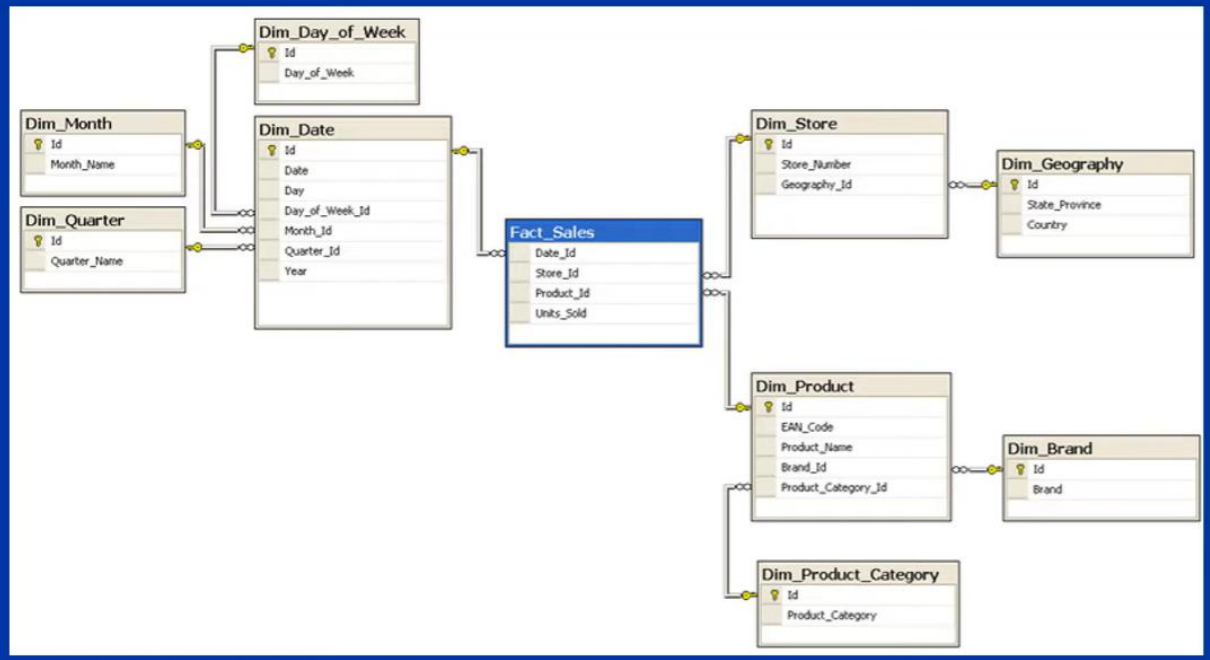
Order Number	Order LineNumber	Customer FirstName	Customer LastName	Sale Date	Product	Quantity	Price
123	1	Bryan	Jones	2013-01-05	Bike	1	$350.00
124	1	Mary	Smith	2013-03-03	Hat	2	$50.00
333	1	Mark	Marks	2013-04-05	Gloves	1	$25.00

Dimensions

Facts

The Snowflake Schema

The Snowflake Schema



When a dimension relates to another dimension you have a snowflake.

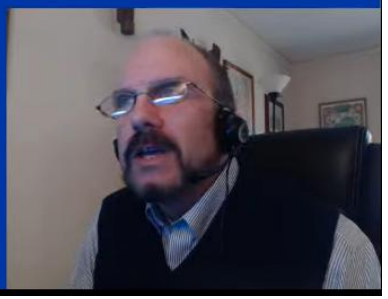
- Beware! OLTP designers must resist the urge to normalize by creating snowflakes.
- Snowflakes cause a number of performance and usability issues and are rarely justified.

Some Key Terms

- Surrogate Keys – artificially created keys (usually integers) used only by the data warehouse to uniquely identify a row in a dimension table.
- Grain – level of detail a fact row represents. For example, sale amount of a single item at a given date/time by salesperson A in the Boston store.
- Conformed Dimension – Different source systems (CRM versus Sales) often have differences in the list of dimension values they support. The CRM system may not have closed branches but the sales system does. A consolidated list of dimension values that supports all the source systems values is called a conformed dimension. Conformed dimensions are critical for a successful data warehouse.

Why Surrogate Keys?

- Required to implement history of slowly changing dimensions.
- Avoids conflicts among backend application keys.
- Insulates the data warehouse from backend application changes.
- Different backend applications may use different columns as the dimension key.
- Note: Typically a surrogate key is just an integer.



Steps of Dimensional Modeling

1. Choose the business process
2. Declare the grain
3. Identify the dimensions
4. Identify the facts

Steps of Dimensional Modeling

1. Choose the business process

The basics in the design build on the actual business process which the data warehouse should cover. Therefore the first step in the model is to describe the business process which the model builds on. This could for instance be a sales situation in a retail store. To describe the business process, one can choose to do this in plain text or use basic Business Process Modeling Notation (BPMN) or other design guides like the Unified Modeling Language (UML).

Steps of Dimensional Modeling

2. Declare the grain

The grain of the model is the exact description of what the dimensional model should be focusing on. This could for instance be "An individual line item on a customer slip from a retail store". To clarify what the grain means, you should pick the central process and describe it with one sentence. Furthermore the grain (sentence) is what you are going to build your dimensions and fact table from. You might find it necessary to go back to this step to alter the grain due to new information gained on what your model is supposed to be able to deliver.
Identify the dimensions

Steps of Dimensional Modeling

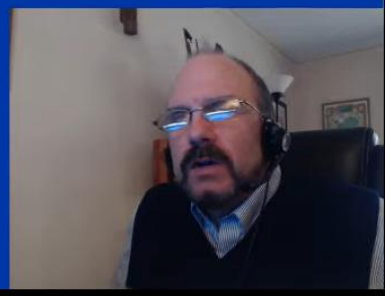
3. Identify the dimensions

The dimensions must be defined within the grain from the second step of the 4-step process. Dimensions are the foundation of the fact table, and is where the data for the fact table is collected. Typically dimensions are nouns like date, store, inventory etc. These dimensions are where all the data is stored. For example, the date dimension could contain data such as year, month and weekday.

Steps of Dimensional Modeling

4. Identify the facts

After defining the dimensions, the next step in the process is to make keys for the fact table. This step is to identify the numeric facts that will populate each fact table row. This step is closely related to the business users of the system, since this is where they get access to data stored in the data warehouse. Therefore most of the fact table rows are numerical, additive figures such as quantity or cost per unit, etc.

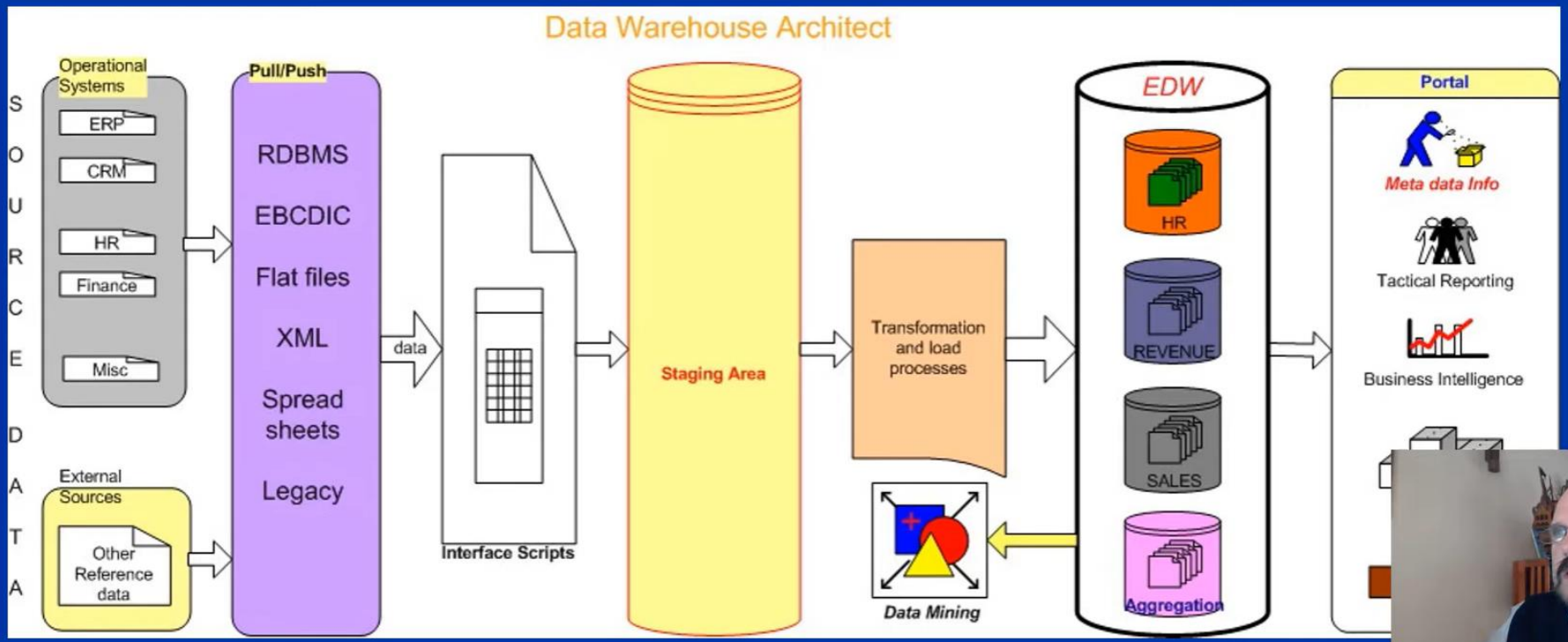


Steps of Dimensional Modeling Review

1. Choose the business process
2. Declare the grain
3. Identify the dimensions
4. Identify the fact

Overview of a Data Warehouse Architecture

Overview of a Data Warehouse Architecture



The Bus Matrix

The Bus Matrix – Finding the Common Dimensions

BUSINESS PROCESSES	COMMON DIMENSIONS							
	Date	Product	Store	Promotion	Warehouse	Vendor	Contract	Shipper
Retail Sales	X	X	X	X				
Retail Inventory	X	X	X					
Retail Deliveries	X	X	X					
Warehouse Inventory	X	X			X	X		
Warehouse Deliveries	X	X			X	X		
Purchase Orders	X	X			X	X	X	X

Figure 3.8 Sample data warehouse bus matrix.

How do I determine what my dimensions and facts are?

The Seven W's of Data Warehouse Design – User Story

The User Story –Describe one instance of the event

Mary Jones buys 1 book for $22.50 entitled "Agile Data Warehouse Design" on December 2, 2013 at 3:12 PM via Amazon.com using her Visa card to be delivered on December 10, 2013 by UPS.

Taken from Agile Data Warehouse Design by Lawrence Corr, DecisionOne Press

Seven W's of Data Warehouse Design and the User Story

The Seven W's of Data Warehouse Design and the User Story

- How?
- What ?
- When?
- Where?
- Who?
- How Many?
- Why?

Taken from Agile Data Warehouse Design by Lawrence Corr, DecisionOne Press

The Seven W's of Data Warehouse Design – User Story

The Seven W's of Data Warehouse Design – User Story

Who?

What

How much? (Fact)

Mary Jones buys 1 book for $22.50 entitled "Agile Data Warehouse Design" on December 2, 2013 at 3:12 PM via Amazon.com using her Visa card to be delivered on December 10, 2013 by UPS.

When?

Where?

How?

Concept from Agile Data Warehouse Design by Lawrence Corr, DecisionOne Press

The Seven W's of Data Warehouse Design – User Story

Who?

What

How much? (Fact)

Boston University orders 2,000 books for $10,000.00 using a Corporate Amazon Account Membership discount with Amazon credit financed at 8% interest.

When?

Where?

How?

Taken from Agile Data Warehouse Design by Lawrence Corr, DecisionOne Press

The Seven W's of Data Warehouse Design – User Story

- Intuitive and natural for business users.
- Efficient way to get the required details.
- Provides jumping off point to get other information such as "Mary ordered via the internet. Are there other outlets for buy products?" or "Mary is an individual, do you have groups or corporate customers?"
- Helps you to focus on a single process at a time.

Concepts from Agile Data Warehouse Design by Lawrence Corr, DecisionOne Press

Slowly Changing Dimensions

What happens when Dimension Values change?

Slowly Changing Dimensions

Dimension values can change and how the change is handled depends on the value the business places on knowing the historical values of a dimension.

Example: The New York store is reassigned from the Northeast Region to the Middle Region. Does management want to be able to see changes in sales as the Northeast Region before the change and Middle Region after the change?

Slowly Changing Dimension – Type 1

Simply overwrite the existing dimension data with the new information.

Advantage: Easiest to implement.

Disadvantage: Lose ability to see how data looked previously.

Example: After the change, all sales history for the New York store would be reported as the Mic

Slowly Changing Dimension – Type 2

We keep all historical values of the dimension.

Advantage: Better ability to report accurately historically.

Disadvantage: Most complex to implement.

Example: Sales in the New York store made prior to the change will be reported in the Northeast Region but any sales after that will be reported in the Middle Region.

Slowly Changing Dimension – Type 3

We keep the prior value and the new value.

Advantage: Easier to implement than SCD Type 3 while still providing some support for historical reporting.

Disadvantage: Historical reporting is limited.

Example: Sales in the New York store made prior to the change will be reported in the Northeast Region but any sales after that will be reported in the Middle Region. However, if the store was moved to the Southeast Region subsequently, you would lose that it was ever in the Northeast Reg

The Date Dimension

Typically uses the date value in integer form such as 20130101 as its key.

Is used to provide descriptive date information, i.e. month name, quarter, day of week name, etc.

SSAS has a wizard to generate this dimension data.

Often has multiple keys in the fact table that point to it. These are called role playing dimensions.

Load with all past and future dates possible from the data.

Review

Dimensional Modeling

Facts (metrics) and Dimensions

Key Terms: Surrogate Key, Grain, Conformed Dimension, Bus Matrix

The Star Schema

Snow flaking Dimensions

Slowly Changing Dimensions

The Date Dimension