BUSINESS DESCRPTION:

NEW ISTANBUL SUPERMARKET CHAIN PLANNING

THE “SUPREMESTORES” COMPANY

BUSINESS DESCRIPTION

BUSINESS BACKGROUND

Shopping malls and superstores have already become an essential part of our everyday life. Here in Istanbul, there are dozens of them, offering a staggering amount of goods and entertainment programs. Owning a franchise of stores like this is one hell of a competitive business, so if you want to be successful and gain an advantage in a business rivalry, you should approach very meticulously to the data you got at hand. Here at "SupremeStores", we firmly believe that an expert team of data analysts & engineers can provide us with all the tools & insights we need to get ahead and push our business to a highly competitive level and secure bigger profits.

PROBLEMS BECAUSE OF POOR DATA MANAGEMENT

Poor data management is equal to losing that aforementioned advantage. If you don't use it – you lose it. Other businesses will definitely not pass on the chance to extract the insightful information from the data available, which will inevitably lead to unrealized profits and even losses. Nowadays, having a firm grip on data is a necessity to stay afloat!

BENEFITS FROM IMPLEMENTING A DATA WAREHOUSE

Now onto more substantial side of things.

One of the ways to strengthen the grip on data is to use a data warehouse. It will enforce data consistency and data veracity, and provide great opportunities for data analysis. The data security will also benefit from having a data warehouse at the core of data operations.

Here are some of the anticipated questions that can be covered by implementing a data warehouse in case of our business:

- The most underrated districts of Istanbul to build a new shopping mall
- The most bang-for-your-buck product categories & products to sell by district
- Promotions & Marketing – push the coupons & discounts setup to be more efficient

Further processing of the data would also allow us to analyze:

- Correlation of customer data with district & product category
- The demand for ATMs in the shopping malls
- Preferred payment method by age & location
- And many others insights.

DATASETS DESCRIPTION

The grain of the dataset is an invoice – a single transaction between client and one of the shopping malls, that represents a single unique tuple in the hypothetical fact table. Also, this grain is already present in a bigger dataset and choosing it will not raise any problems in merging both sources together.

The granularity of time dimension (i.e. its resolution) is seconds. However, this resolution is way too high for a data warehouse – "minutes" resolution will be implemented.

The detailed description of the present dimensions & facts:

(Notice that the naming conventions are not being followed as of yet. DIM tables are not normalized.)

Dimensions Description:

Customers dimension:

Stores all the customer-related information, apart from customer ratings, which is a different entity.

Column name	Description	Data Type
Customer_id	Unique customer id, PK	Int
Customer_name	Full customer name	Text
Customer_gender	Female/Male	Text
Customer_age	Customer's age	Text

Example with filled data:

Customer_id	Customer_name	Customer_gender	Customer_age
1	Jim Morrison	Male	79

Time dimension:

Time dimension is a special, calendar-like dimension that stores all the dates present in the data sources with a certain granularity: a "day" in our case. Hours and Minutes are separated and will be depicted in the "Sales" fact table. The rest of the columns are going to be "event_dt" derivatives, like day of week, quarters, fiscal/calendar month, year, etc, to improve data analytics experience.

Column name	Description	Data Type
Event_dt	Unique Date, PK	Date
Day_of_Week	Time_id's day of week	Text
Calendar_Week_Num	Week number within a year	Int
…rest of the calendar columns	…	…

Example with filled data:

Event_dt	Calendar_Week_Num	Day_of_Week	…
12/12/2022	52	Wednesday	…

Product category dimension:

This dataset is not concerned about particular products, rather only their categories. The `category` attribute is present in the table and can be seen as a dimension.

Column name	Description	Data Type
Category_id	Unique category Id, PK	Int
Category_name	Product Category	Text

Example with filled data:

Category_id	Category_name
1	Souvenir

Stores dimension:

The stores dimension consists of store information, including company-owner and location in Istanbul.

Column name	Description	Data Type
Store_id	Unique store id, PK	Int
Store_name	Name of the store	Text
Company_name	Owner company name	Text
District	Istanbul District	Text
Lat	Latitude	Float
Long	Longitude	Float

Example with filled data:

Store_id	Store_name	Company_name	District	Lat	Long
1	Metrocity	AnkorTech	Fatih	41.004582	28.863052

Discount dimension:

This dimension stores information on coupons and the corresponding discounts.

Column name	Description	Data Type
Coupon_id	Unique coupon id, PK	Int
Discount_size	Size of the discount, in %	Text

Example with filled data:

Coupon_id	Discount_size
-1	0%

Facts Description

Sales Fact:

Sales fact table appears in a bigger dataset and depicts all sales with a natural key `invoice_no`. Keeping in mind all the dimensions, and all the additional numeric characteristics like costs & revenue, it would look like this:

Column name	Description	Data Type
Event_dt,	Unique time id, FK	Date
Event_minutes	Time of day (transaction time)	Timestamp
Invoice_no (Sale_id)	Natural key for the sales, PK	Int
Customer_id	Unique customer id, FK	Int
Store_id	Unique store id, FK	Int
Category_id	Unique category id, FK	Int
Coupon_id	Unique coupon id, FK	Int
Survey_id	Unique survey id, FK	Int
Payment_method_id	Either Cash or Cards	Int
Quantity	Number of goods purchased	Int
Price	Price per 1 item	Float
Costs	Amount of costs per 1 item	Float
Discount	Total Discount per Invoice	Float
Revenue	Revenue per Invoice	Float
Payment_amount	Amount paid by customer	Float

Example with filled data:

Event_dt	Event_minutes	Invoice_no	Customer_id	Store_id	Category_id	Coupon_id
12/12/2022	14:23:56	I013234	C103292	1	2	-1

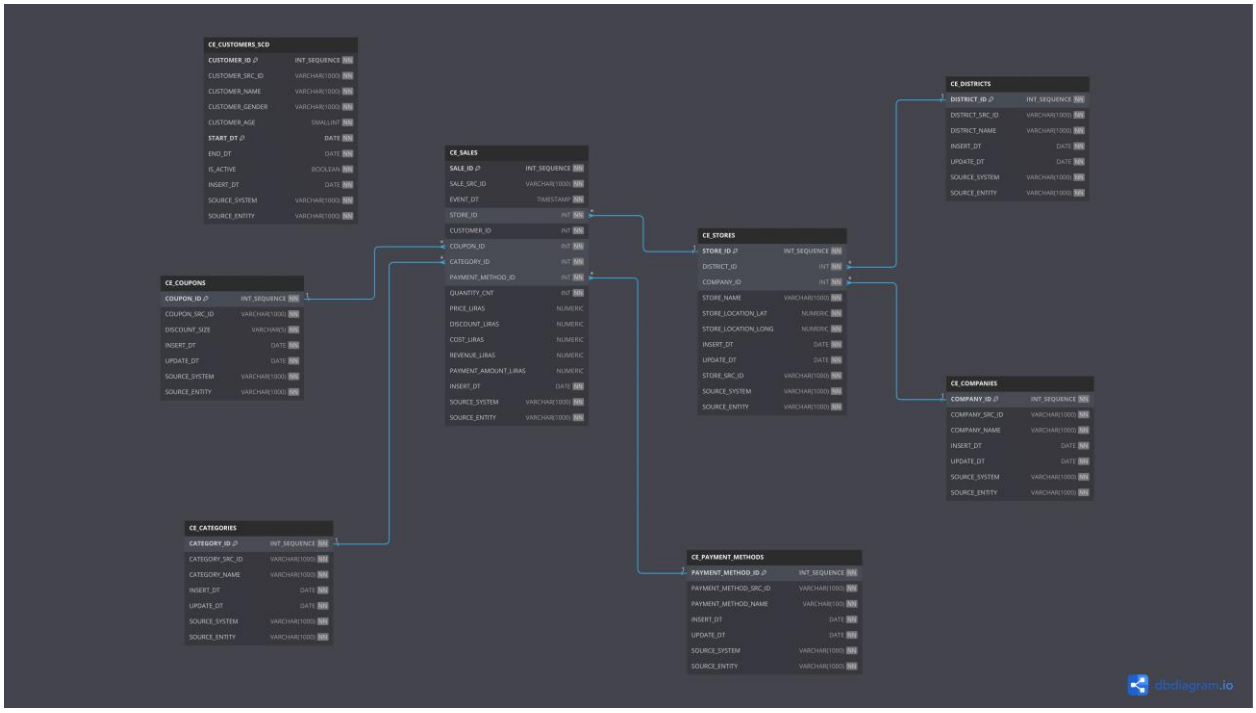
Survey_id	Payment_method_id	Quantity	Price	Costs	Discount	Revenue	Payment_amount
-1	1	3	3.0	2.0	0.0	3.0	9.0

BUSINESS LAYER RELATIONAL MODEL

The process of building a relational model (later – 3NF model) can be divided into 3 stages/layers of abstraction:

1. Conceptual model
2. Logical model
3. Physical model

But firstly, we need to combine all the data sets from all the sources into one pool of entities: If an entity is present in at least one data source – it is being included into 3NF model ('OR' logic).
After that, we strictly follow the process of building these layers. From establishing entity relationships and finding their corresponding key attributes, to normalization and physical nuances like data types, NULL policies and technical attributes. The resultant BL_3NF model is the following:

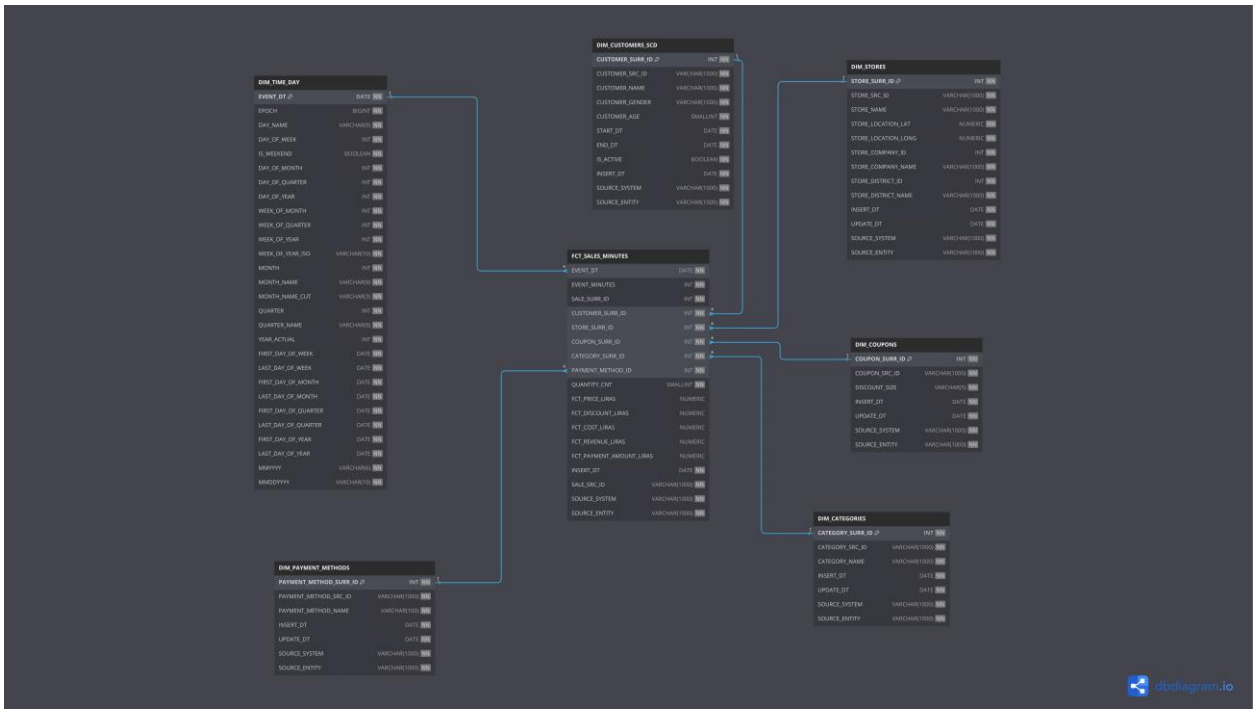


Note: "Customer_id" column is a logical foreign key, meaning that the relationship between "Customers" entity and "Sales" is respected, however not on physical level, because Survey's primary key is composite (SCD-2 table).

BUSINESS LAYER DIMENSIONAL MODEL

Having already established BL_3NF model, we can now transform it into a Star Schema by following the 4-step process of dimensional data modeling. We are also required to strictly follow naming conventions that are aimed at easing the process of analyzing data.

The resultant Star schema is the following:



The fact table consists of multiple measurements regarding the transaction, such as:

- FCT_QUANTITY_CNT:

  Number of items per transaction (sale)

- FCT_PRICE_LIRAS:

  Price for a single item in transaction (sale) in Turkish Liras

- FCT_DISCOUNT_LIRAS:

  Discount size for the whole transaction (sale) in Turkish Liras

- FCT_COST_LIRAS:

  Cost of a single item in transaction (sale) in Turkish Liras

- FCT_REVENUE_LIRAS:

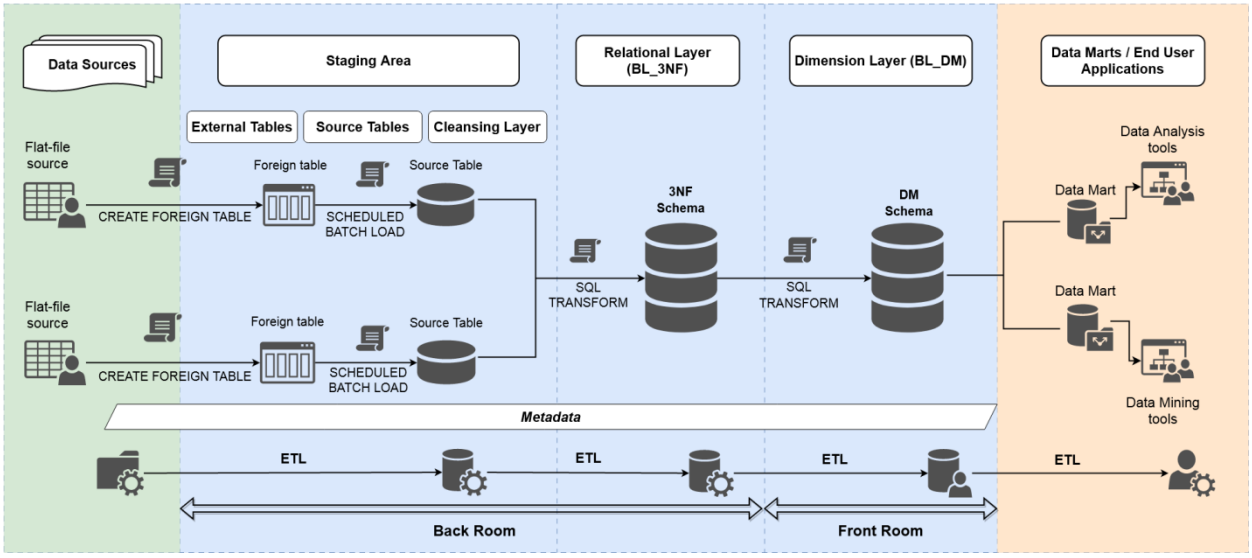
  Revenue from a whole transaction (sale) in Turkish Liras:

- FCT_PAYMENT_AMOUNT_LIRAS:

  Amount customer paid for a transaction(s). For card payments only.

LOGICAL SCHEME

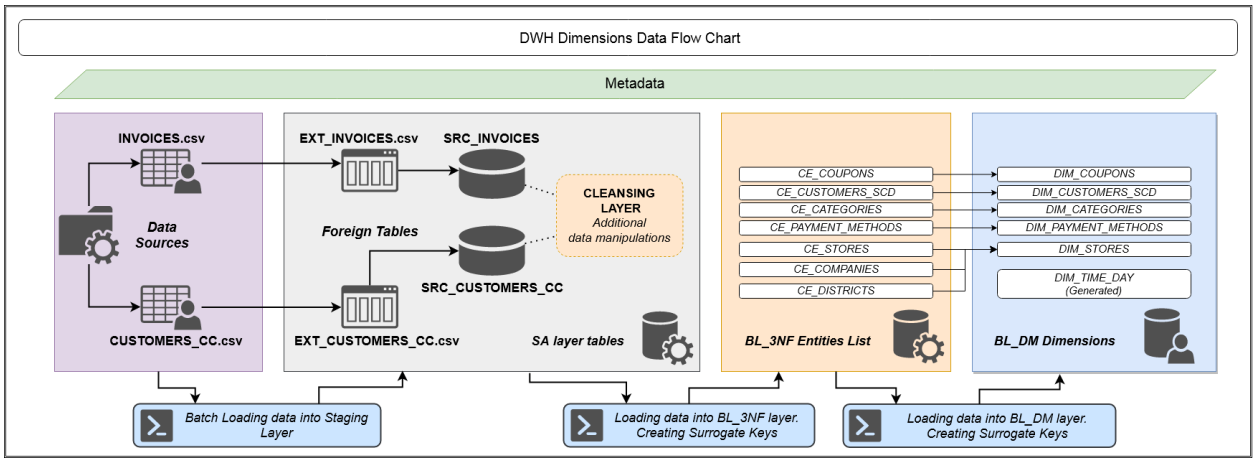
The logical scheme represents the data warehouse architecture in terms of loading process. It demonstrates the path of the source data to the end-user inside of the data warehouse, depicting all the inner DWH layers of data representation.



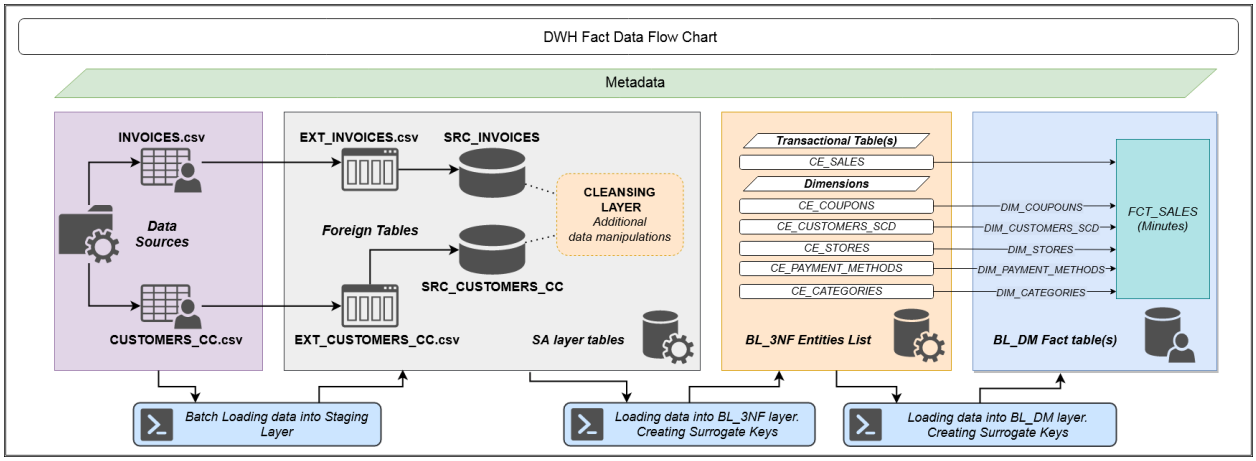
DATA FLOW

The data flow chart represents a more detailed approach to tracing source data in the DWH. Its granularity is dimensions and entities, unlike the logical schema. In order to achieve a comprehensive schema, original data flow diagram was divided into two: for dimensions and for “SALES” fact table.

Dimensions Data Flow:



Facts Data Flow:



FACT TABLE PARTITIONING STRATEGY

Fact Table Partitioning Strategy:

Range partitioning by `event_dt` column

—

Each partition represent a half-year period.