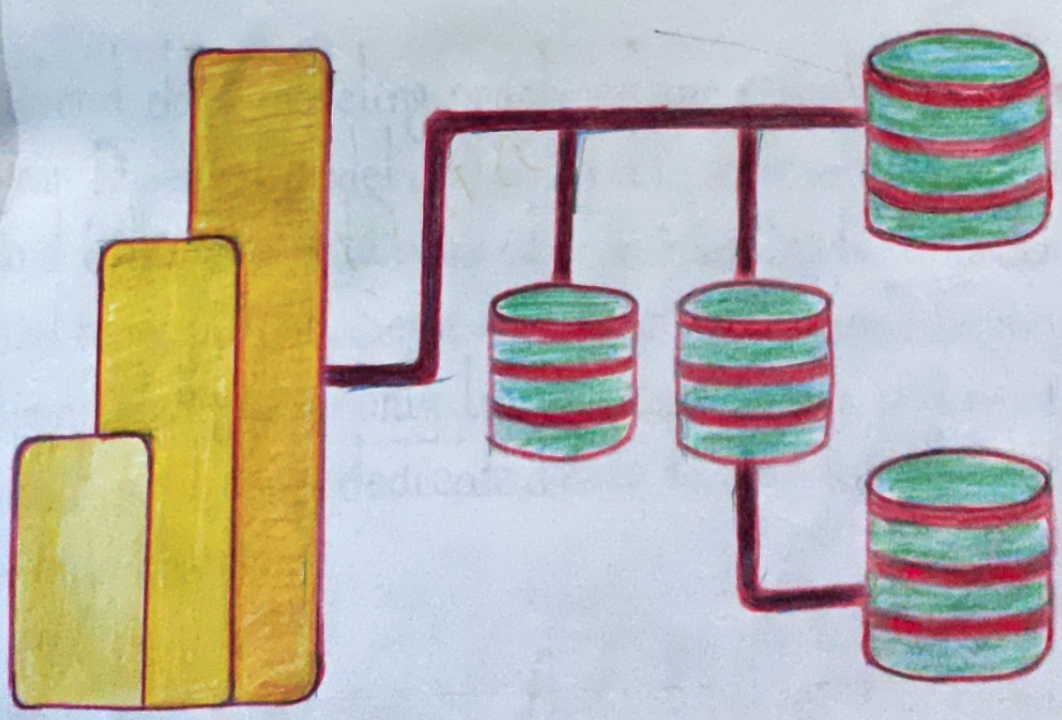
Data Modeling Fundamentals in Power BI

Data modeling in Power BI involves creating a data model that defines the data structure, properties, and relationships within a dataset. Power BI automa-tically detects relationships between different sets of data, but users can also create custom relationships manually. The model view in Power BI displays the data model, which looks similar to an entity relationship diagram, with tables and their corresponding columns, as well as relationships between them.



To create a data model in Power BI,

1. Import data from various sources, such as an Excel sheet, using the 'Get Data' option in the 'Home' menu.
2. After the data import is complete, visit the 'Models' tab from the left side.
3. Here, you will see automated relationships created by Power BI. You can create and modify these relationships using the 'Manage Relationship' tool given on the top.
4. After clicking on 'Manage Relationship', a similar screen will appear, allowing you to manage the relationships between tables.

Good data modeling practices are essential for optimizing your Power BI model, allowing it to scale into the millions and billions of data rows while maintaining fast rendering and refresh time. Some best practices include turning off time intelligence, only loading the columns and rows you need, and using dedicated date tables for all date/time hierarchies.

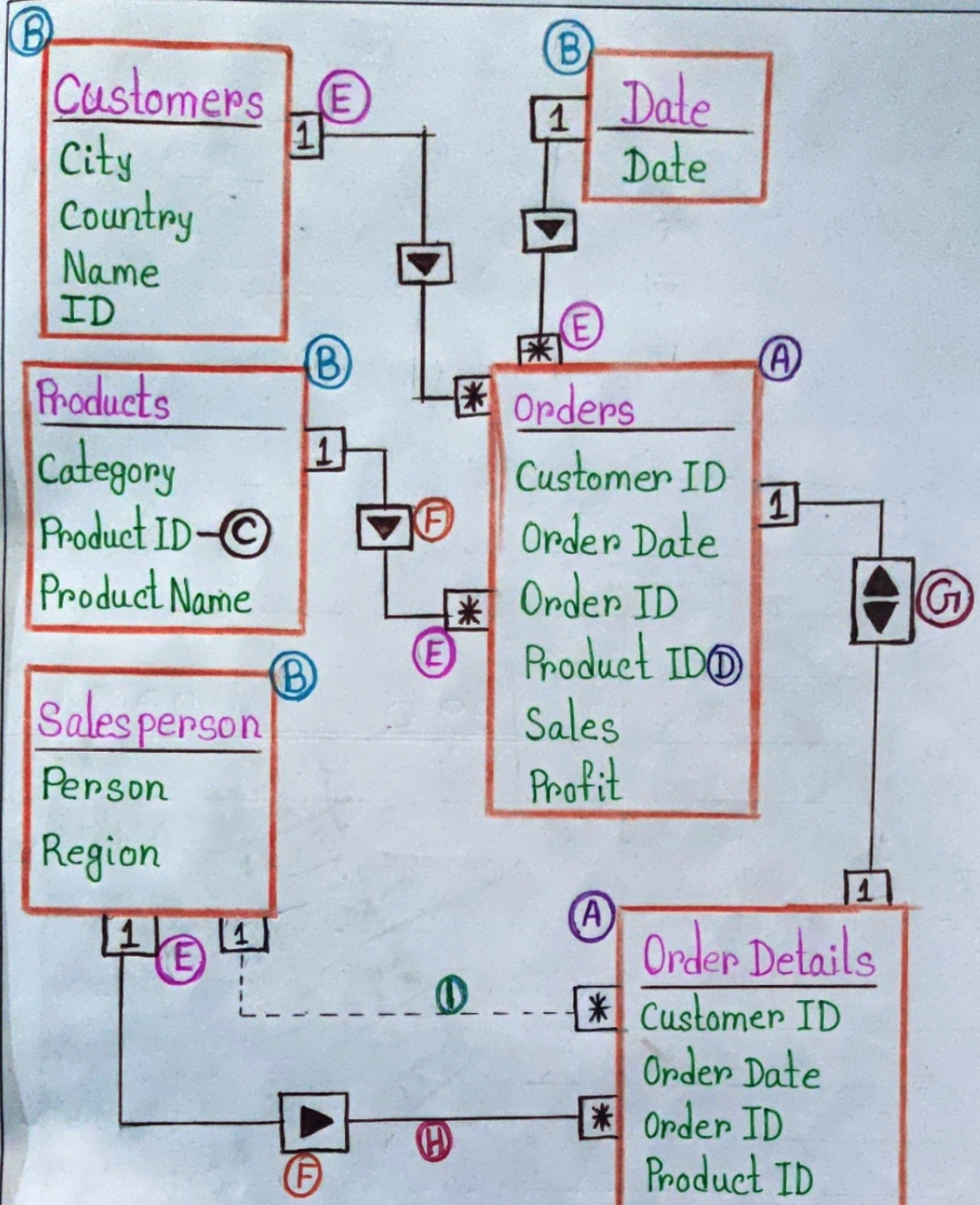
Fact Tables A fact tables stores quantitative, numerical data that can be aggregated and

analyzed in different ways. It typically contains information about specific events, transactions, or measurements and serves as the centerpiece of a data model. In a fact table, each row represents a single transaction or event, and each column represents a measurable attribute or fact about that event. For example, a sales fact table might have columns for the date of the sale, the product sold, the quantity sold, and the revenue generated.

Dimension Tables

A dimension table provides context and additional details to the numerical data stored in a fact table. For example, a product dimension table might contain columns for the product name, product category, product description, and supplier name.

Generally, dimension tables contain a relatively small number of rows. Fact tables, on the other hand, can hold a very large number of rows and continue to grow over time. In a data model, a dimension table is linked to a fact table through a relationship. By linking the two tables together, Power BI users can analyze the data in the fact table by filtering, grouping, and aggregating the data based on the attributes in the dimension table.



Ⓐ- FACT TABLE
Ⓑ- DIMENSION TABLE
Ⓒ- PRIMARY KEY
Ⓓ- ALTERNATIVE KEY
Ⓔ- CARDINALITY
Ⓕ- SINGLE DIRECTIONAL CROSS FILTER
Ⓖ- BI-DIRECTIONAL CROSS FILTER
Ⓗ-ACTIVE RELATIONSHIP
Ⓘ- INACTIVE RELATIONSHIP

ER DIAGRAM

ER Diagram

An entity relationship (ER) diagram is a visual representation of the relationships between different entities in a database. It is used to design and understand the stucture of a database by illustrating the entities, their attributes, and the relation--ships between them. In Power BI, data models can be created to resemble ER diagrams, with tables represen--ting entities and relationships between them represen--ting the connections between these entities.

Relationship Fundamental Concepts

Primary and Alternative Key

To create a relationship between two tables that need to share a column in common, that's where primary and alternative keys come in. Primary and alternative keys are columns in a table that create a relationship between two tables.

Primary Key

Also known as a surrogate key, a primary key is a column or group of columns in a table whose values uniquely identify a row in the table. The primary key's distinct count equals the number of rows in the table.

Alternative Key

Also known as a foreign key, an alternative key is a column in a table whose values correspond to the values of a primary key in another table.

In the previous diagram example, in the Products dimension table, the primary key is the Product ID column, which uniquely identifies a product or row in the Products table. The Products dimension table is connected to the Orders fact table based on Product ID, so the Product ID column within the Orders table would act as an alternative key.

Cardinality

Our fact and dimension tables connect via primary and alternative keys, but we must define their relationship further. Each relationship in a model is defined by a cardinality type. It refers to the number of unique values in one table related to the number of unique values in another.

There are three types of cardinality relationships in Power BI data modeling.

One-to-One (1:1)

This occurs when one record in the first table is related to one and only record in the second table. This type of relationship is relatively rare in Power BI data modeling.

One-to-Many (1:N)

This occurs when one record in the first table can be related to many records in the second table, and it is the most common type of relationship in Power BI data modeling. This cardinality type is used to link a fact table with one or more dimension tables, where the dimension table is typically on the "one" side of the relationship, and the fact table generally is on the "many" side.

Many-to-Many (N:N)

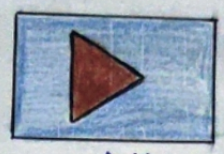
This occurs when many records in the first table can be related to many records in the second table. This relationship type is not directly supported in Power BI data modeling and is infrequently used.

Cross Filter Direction

Each relationship in a model is also defined with a cross-filter direction, which determines the direction(s) that filters will propagate. The cross filter direction can be either single,

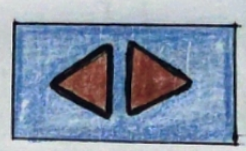
which we can think of as "Single direction", or both, which we can think of as "both directions" or bi-directional.

• Single



When a single-directional cross filter is applied, only one of the tables in the relationship can filter data across to the other table. This means filtering data in one table affects the data shown in the other table, but filtering in the other table does not affect the data shown in the first table.

• Bi-Directional



When a bi-directional cross filter is applied, data can be filtered in either direction across the relationship between the tables. This means filtering data in one table affects the data shown in the other table and vice versa. This type is less common and can negatively impact performance and create ambiguous filter paths.

The possible cross-filter options are dependent on the cardinality type. As shown in the next table, relationships with a 1:1 cardinality can only have a bi-directional cross-filter direction, whereas 1:Many and Many:Many relationships can have either single or bi-directional cross-filter directions.

Cardinality	Cross Filter Options
1:1	Bi-Directional
1:Many	Single or Bi-Directional
Many:Many	Single or Bi-Directional

Active and Inactive Relationships

By default, when a relationship is created between two tables, it is considered an active relationship. This means the tables are joined based on that relationship, and the related tables are filtered by the active relationship. However, it is possible to have multiple relationships between two tables, and in some cases, having more than one relatio--nship between tables may be necessary. This is where active and inactive relationships come into play.

Between two tables, you can only have one active relationship at a time to avoid conflicting or incorrect results. So when multiple relationships exist between two tables, some must be inactive.