



★ Get unlimited access to the best of Medium for less than \$1/week. [Become a member](#)



Data Modeling — Basics



Chayan Shrang Raj · [Follow](#)

6 min read · Nov 15



284



When developing a database system, we need to make sure that it operates efficiently and that we can extract information from it quickly. The best way to create such a system is to first design a data model. With a data model, you can plan how data is stored and accessed within your database **before** you create the database system.

Data Modeling is a process of developing visual representation of data elements and their relationships. It tells us how our data is structured within the database. This structure helps us to understand how data is stored, accessed, updated, and queried within the database.

The goal is to illustrate the types of data used and stored within the system, the relationships among these data types, the ways the data can be grouped and organized and its formats and attributes.

Importance of Data Modeling — Data models are constructed based on the requirements of the business. Initial rules and specifications are

established through input from business stakeholders, informing the creation of a new system or adjustments to an existing one. Communicating data needs in this step becomes crucial in creating a data plan.

The modeling of data occurs at various levels of abstraction, starting with the gathering of business requirement details from stakeholders and end users. These business rules are then translated into data structures to develop a tangible database design. Comparatively, a data model acts as a guide, resembling a roadmap, an architect's blueprint, or any formal diagram that enhances the comprehension of the design.

Standardized schemas and formal techniques are employed in data modeling, ensuring a consistent and predictable approach to defining and managing data resources across an organization or even beyond.

In an ideal scenario, data models are dynamic documents that adapt to changing business needs. They play a crucial role in supporting business processes, as well as in the planning of IT architecture and strategy. Moreover, data models can be shared with vendors, partners, and/or industry peers.

There are three different levels of Data Modeling:

1. Conceptual data models
2. Logical data models
3. Physical data models

Let's discuss each of them in more detail.

1. Conceptual data model — It consists of high level abstract of data elements or entities and their relationship with each other. This can be created by understanding the business process that you want to model. Conceptual models are usually created as part of the process of gathering initial project requirements.

Entities can be defined as real life, independent existence of a thing, For example, Cars, People, Restaurants, Products, etc.

Relationships are defined as the process in which entities interact with each other, For example, Customers place orders, people buy food, etc. Here the the place and buy define the relationship between two entities.

The conceptual model provides the basis for logical data model.

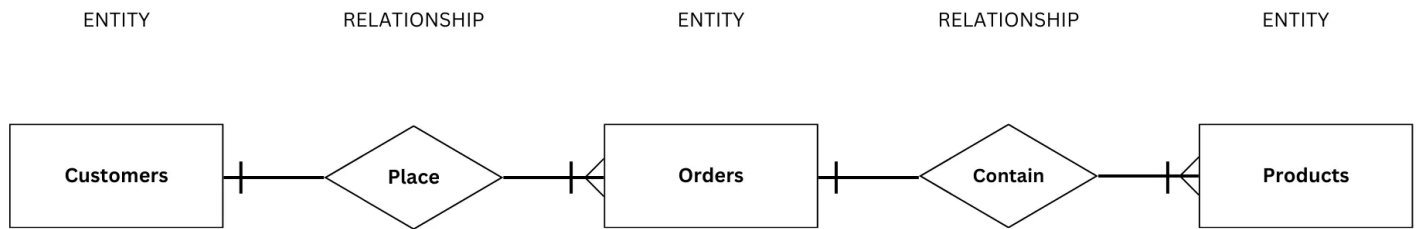
Pros:

- ER diagram presents the database structure in an easy-to-use fashion.
- It is very easy to communicate the business needs with non-technical stakeholders.

Cons:

- It may not be able to capture complex requirements as the business scales.
- It may not represent the final product.

In the figure below, a conceptual data model is shown for an ecommerce website where a customer places one or more orders, where each order may contain one or more products.



Conceptual data model (Image credits: Author)

2. Logical data model — The logical data model builds on top of conceptual model by providing a detailed overview of the entities in their respective relationships. It identifies the attributes of each entity, defines the primary keys, and specifies the foreign keys.

The logical data model must include all attributes required for each entity. Like a list of the attributes each entity contains. It then needs to define which of these columns serve as the primary key for each table and also specify foreign keys that create relationships between each entity.

Pros:

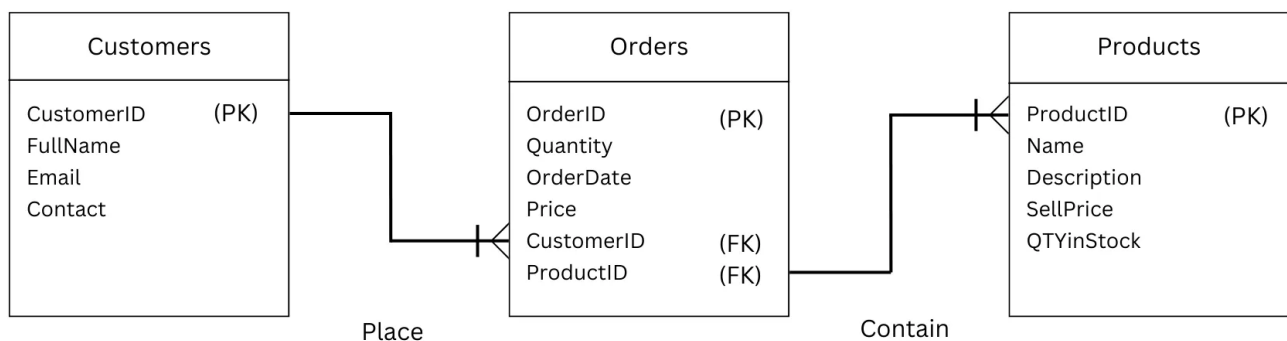
- It contains more information than conceptual model such as entities, attributes, relationships and constraints, so it contains more business information and processes, which brings more clarity.

- A larger fraction of business system requirements can be better expressed in terms of information than can be expressed in terms of processes.

Cons:

- Few people can understand when it is first presented to them because it requires in depth understanding of the business process.
- A highly structured data model can become rigid and less adaptable to evolving business needs as there could be entities that are very tightly linked together.

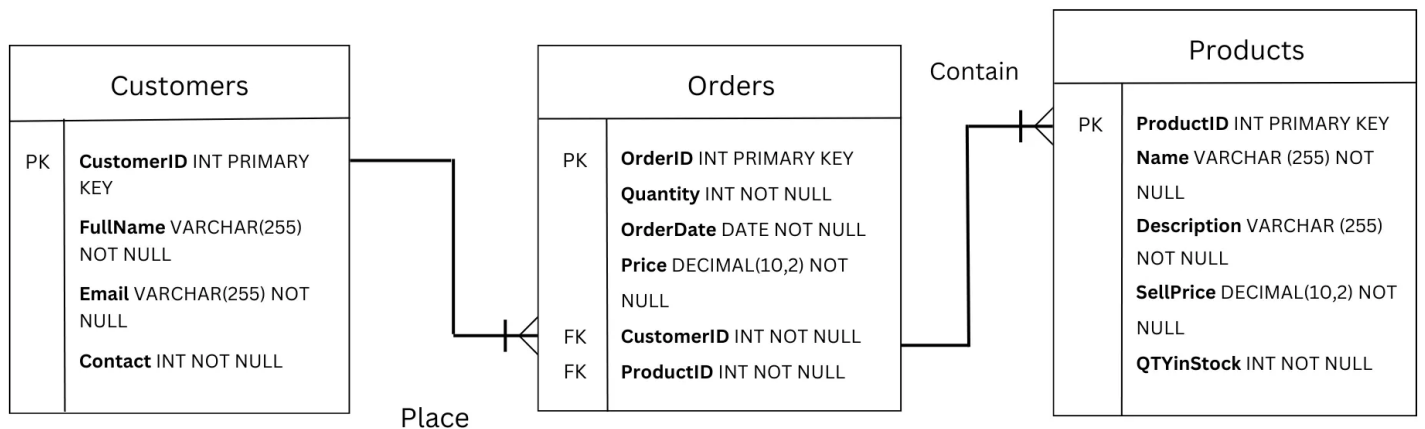
In the figure below, we have three tables (entities) Customers, Orders, Products each with their corresponding primary key and foreign keys. In this model, customers table is connected to orders table through 'CustomerID' foreign key and Orders table is connected to Products table through 'ProductID' foreign key.



3. Physical data model — A physical data model is used to create the internal SQL schema of the database, which is implemented in the database management system. The physical data model must outline features like the datatypes, constraints, and attributes.

They provide a schema for how the data will be physically stored within a database. As such, they're the least abstract of all. It changes the question of 'what' to model to 'how' to model data.

In the figure below, each attribute has a data type. It also needs to have relevant constraints. A constraint of NOT NULL for each column in the customer's table makes sure that each one contains data. Primary keys denote that each row of that table will be uniquely defined.



Before, we discussed what incremental steps should we take to craft a data model from scratch and build our database system. But, it would be unwise to think that our world is contracted to only one or two data models, in fact, there are many different kinds of data models such as:

a) Relational data model — It's a popular and widely used database model. It represents the database as a collection of relations. Each relation is presented as a table that stores information in the form of rows and columns. A key advantage of this model is that it's much simpler to use than other models. You can quickly identify and access data. But the relationships between the data in this model can become more difficult to navigate with complex relational database systems.

b) Entity relationship model — This model is similar to the relational data model. The key difference is that you can present each table as a separate entity by assigning each one its own set of attributes. The model also covers many different types of relationships between entities such as one-to-one, one-to-many, and many-to-many relationships.

c) Hierarchical data model — The hierarchical data model organizes data in a tree-like or parent-child structure. Each record of data has a parent node, and it can also have its own child node. The main disadvantage is that it can only be used to record one-to-many relationships between nodes. Each child node can only have one parent node.

d) Object-oriented model — Another option for database developers is the object oriented model. This model is based on the object oriented concept. This is where each object is translated to a class that defines the objects characteristics and behavior. A key advantage of this model is that you can define different types of associations between objects, like aggregations,

compositions, and inheritance. This makes object oriented databases suitable for complicated projects that require an object oriented approach.

e) Dimensional data model — This model is based on two key concepts, dimensions and facts. Facts are measurements obtained from a process. For example, sales facts obtained from ecommerce business data. Dimensions define the context of these measurements, like a specific sales period. Sales facts measure how many quantities of a particular product is sold each week. The key advantage of this model is that it optimizes the database for faster data retrieval and restructures data for more efficient data analytics.

We will discuss more about them in a different article.

Now, there are many different ways to build data models using both traditional and modern data modeling tools. It depends on the organization, cost, departments, stakeholder collaboration, and many other factors in order to get out of decision overload and choose the most effective solution.

Some examples are MySQL Workbench, ER/Studio, Archi, etc..

Next, we will dive deeper into relational and dimensional modeling, two of the most common and widely adopted data models.

Thanks for reading!

Github: <https://github.com/chayansraj>

Data

Data Modeling

Data Engineering

Data Science

Sql