Widgets - System description and database schema

Paweł Rubin

September 2020

Introduction

The task is to create a basic system description and document a normalized schema from the attached widgets text file.

- what you think this system would do
- what you feel would be a reasonable database structure for the data and a reasonable architecture for the system
- any questions or concerns you have regarding this dataset/system that might need to be answered before establishing an ideal database/solution for such a system.

It's a very open-ended problem, and that's part of the problem.

Analysis of the data file

Based on the provided data file, we can assume that the system is responsible for documenting some kind of transactions. It could be an online store or a warehouse with various widgets.

I think it can be safely assumed that **widgets.tsv** is a list of orders.

We can distinguish the following entities:

- **Widget** - Our products
- **Customer** - Set of customers
- **Supplier** - List of suppliers
- **Warehouse** - Our warehouses - could be an enum if we don't need details
- **Packaging** - Types of packaging - also could be an enum

Different price for different customer

It should be noted that each of **Elephant Trap**, **Ant Trap** or **Moose Trap** has a different price depending on a customer although the other parameters are the same.

Has the price changed over time or it depends on the order?

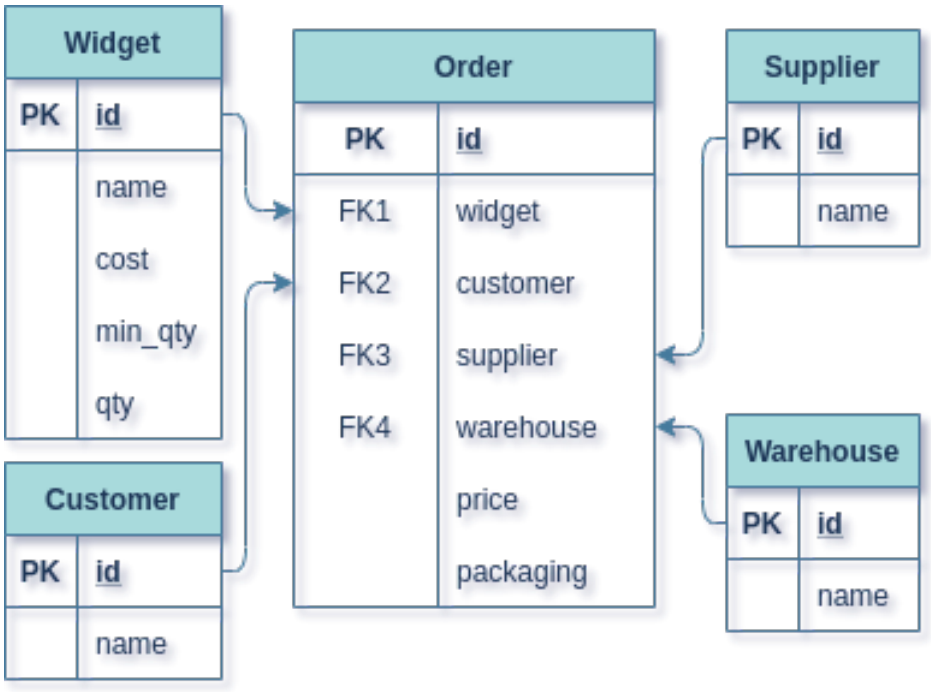

Figure 1: Database schema

Quantity

There are two columns regarding *quantity*: **qty** and **min_qty**. The latter probably means what is the minimal quantity to place an order on particular item. **qty** however could mean how many items left in storage since **qty** times **cost** is significantly larger than the **price**.

Data sort order

Data is presumably ordered by **min_qty** value.

What is the reason? Is it to parse orders, that most likely fit the condition, first?

Database schema

Figure 1 shows basic schema of possible database.

Orders should also contain a timestamp and a cost (depending on business logic and if it's changing over time). Packaging types could also be stored in the database and be correlated with each widget.

Possible differences based on business logic

- **price** could change over time, depends on a client or both. It possibly includes shipping costs, depending on warehouse localization.
- **min_qty** could also differ over time - for example if cost changes.
- Widgets details could also be stored in the **Order** collection - depending on the logic above. If widgets parameters (cost, min_qty) changes over time - we want to store information from the time of the transaction.
- Is it an in-house system or a public one?

System description

It system could be an in-house system or a widget store opened to the public. However, no matter the purpose, we could develop a web application. Choice of a frontend/backend frameworks is up to developers - for example *Django* could be use to easily implement our data models.