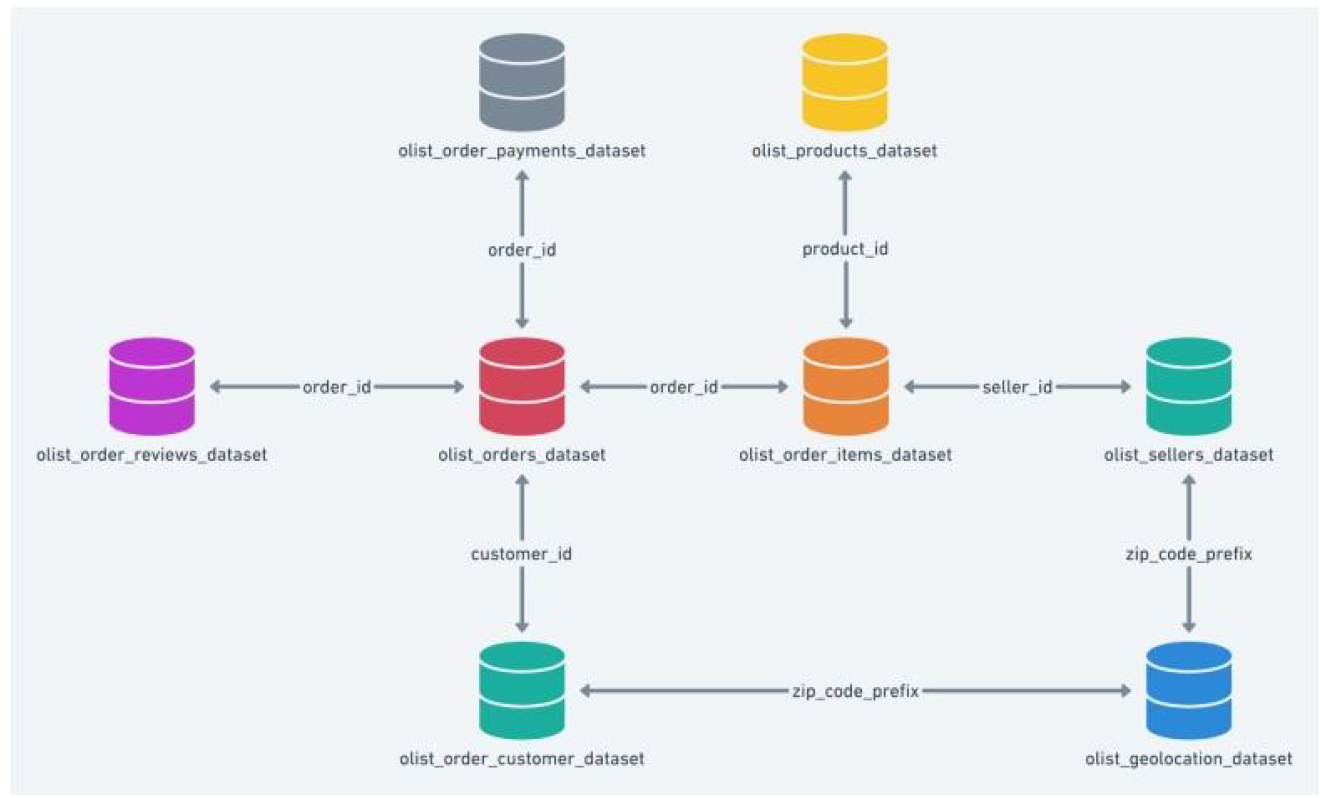
The client told us that he lacks granular information about sales and customers. He needs some metrics and I started some analysis using SQL with what we have in our data warehouse.

This is the ERD:



Before diving into the main query, I want to anticipate you some particular issues that I found:

* **Customer**\_id (string type) is not a unique value. Apparently, each time a customer enters into our client’s market and buys something, the system assigns a random string value despite the fact that he (maybe) made a purchase in the morning, then came back to the market at the evening, and bought again. **Customer\_unique\_id** is the unique value that corresponds to each customer.
* Some **orders\_id** are not present in **orders\_items\_dataset** table, which means that a rate (%) of sales can’t be quantified in monetary ($$) values. We should do some research or bring up the backend team and work around together this issue.
* I picked the **order\_purchase\_timestamp** (Date) as the vector to map this analysis, I thought it would be great to look at the sales growth over days.

The query involves:

1. Total Orders: I count unique values
2. Clients:
3. Revenue:

Loader to Snowflake

Modifications if deployed in Production:

It’s most likely that our client wants to have a service deployed, when I say service, I mean an application(product) that executes several connected instructions and produces some useful output. In order to deliver this kind of service, I’d list some possible upgrades:

* **Full development in OOP**: We could leverage this development with OOP, packetize these functions and convert them into a scalable product. This will serve another project or demands from our client or ANY other client that has the same necessity. And we will do it just once, honoring DRY Concepts from software development (Don’t Repeat Yourself). Moreover, when products are developed in OOP, it’s easier to debug and build in functionalities.
* **Preprocessing, cleaning & normalizing**: Talking about specifically from our application, we should deep dive into data, it’s almost certain that we must do some preprocessing over data. Normalizing or cleaning: filling zeros, null/nan values or empty rows, standardizing values if needed, parsing datetime, casting columns as integers or strings, renaming columns, delete duplicated rows or columns or any another tasks in terms of preprocessing.
* **Keep our product safe**: We could develop a pipeline with solid logic and flow control (i.e.: our product could evaluate file’s modification date or check if preprocessing it’s OK) before deploying to avoid any possible further issues or bugs in production.