Cargo Shipping in Data Warehousing

Introduction: -

Cargo shipping data warehouse is a centralized repository that stores and manages large amounts of data related to the cargo shipping industry. It is a powerful tool that enables businesses to efficiently analyse and make informed decisions based on historical and current data. This data warehouse contains a wide range of information, including but not limited to, shipment details, transportation routes, customs declarations, tariff codes, and other relevant metrics. The warehouse can be used to analyse trends and patterns in shipping data, improve supply chain efficiency, reduce operational costs, and identify potential opportunities for growth. With the increasing demand for international trade and the growing importance of logistics and supply chain management, cargo shipping data warehouses have become an essential asset for companies operating in this industry.

Scenario: -

Mediterranean Shipping Company (MSC), one of the world's leading shipping lines, has decided to invest in a data warehousing solution to improve their business intelligence and decision-making capabilities. The company wants to create a centralized repository of all their data that can be easily accessed and analysed by their business analysts and executives.

To start the project, MSC first needs to identify the key data sources that will be included in the data warehouse. These could include transactional data from their shipping operations, financial data from their accounting systems, customer data from their CRM system, and logistics data from their supply chain systems.

Developing a Data Warehouse: -

- Identify and Collecting Requirements:-

1. Identify data sources for the data warehouse, including shipping operations, financial data, customer data, and supply chain systems.
2. Cleanse and transform data before loading it into the data warehouse.
3. Design a schema that supports key business processes and metrics and enables reporting and analytics.
4. Select a data warehousing platform that is scalable, performs well, integrates with existing systems, and uses MySQL.
5. Build out infrastructure, including hardware and software components, to support the data warehousing platform and integrate data effectively.
6. Design and build reports and dashboards for reporting and analytics that can be accessed by different stakeholders.
7. Implement security measures to ensure data is secure and protected.
8. Optimize performance for indexing, query optimization, and handling large amounts of data.
9. Ensure the data warehouse is scalable for business growth.
10. Provide training and support for users of the data warehouse.

Dimension Table: -

The dimension table in a shipping dimensional model would typically include the following attributes:

- Date: This would include information about the date and time of the shipment, such as the year, month, day, hour, and minute.
- Customer: This would include information about the customer who placed the order, such as their name, address, email address, and phone number.
- Product: This would include information about the product being shipped, such as its name, description, price, and SKU.
- Shipping Method: This would include information about the method of shipping, such as the carrier, service level, and cost.
- Shipping Address: This would include information about the destination address of the shipment, such as the street address, city, state, and zip code.
- Payment: This would include information about the payment for the shipment, such as the payment method, payment amount, and payment status.

These attributes would be used to define the dimension table, which would be related to the fact table by a foreign key. The fact table would contain measures such as the number of items shipped, the total shipping cost, and the total revenue generated from shipping. By using a dimensional model like this, businesses can more easily analyse and understand their shipping data, identify trends, and make data-driven decisions.

Product Dimenssion
Product ID (PK)
Product Name
Brand
Category
Charge
Other Attributes

Customer Type Dimension
Customer Type ID (PK)
Customer Size
Customer Shipping Frequency
Credit Status

City Dimension
City ID (PK)
Sub-Region
Region
Territory
Other Attributes

Date Dimension
Date ID (PK)
Year
Month
Day
Other Attributes

Shipment Dimenssion
Shipping ID (PK)
Shipping Mode
Shipping Class
Other Attributes

Customer Dimension
Customer ID (PK)
Customer Type ID (FK)
Customer Name
Other Attributes

Deal Dimenssion
Deal ID (PK)
Other Attributes

Contract Dimenssion
Contract ID (PK)
Other Attributes

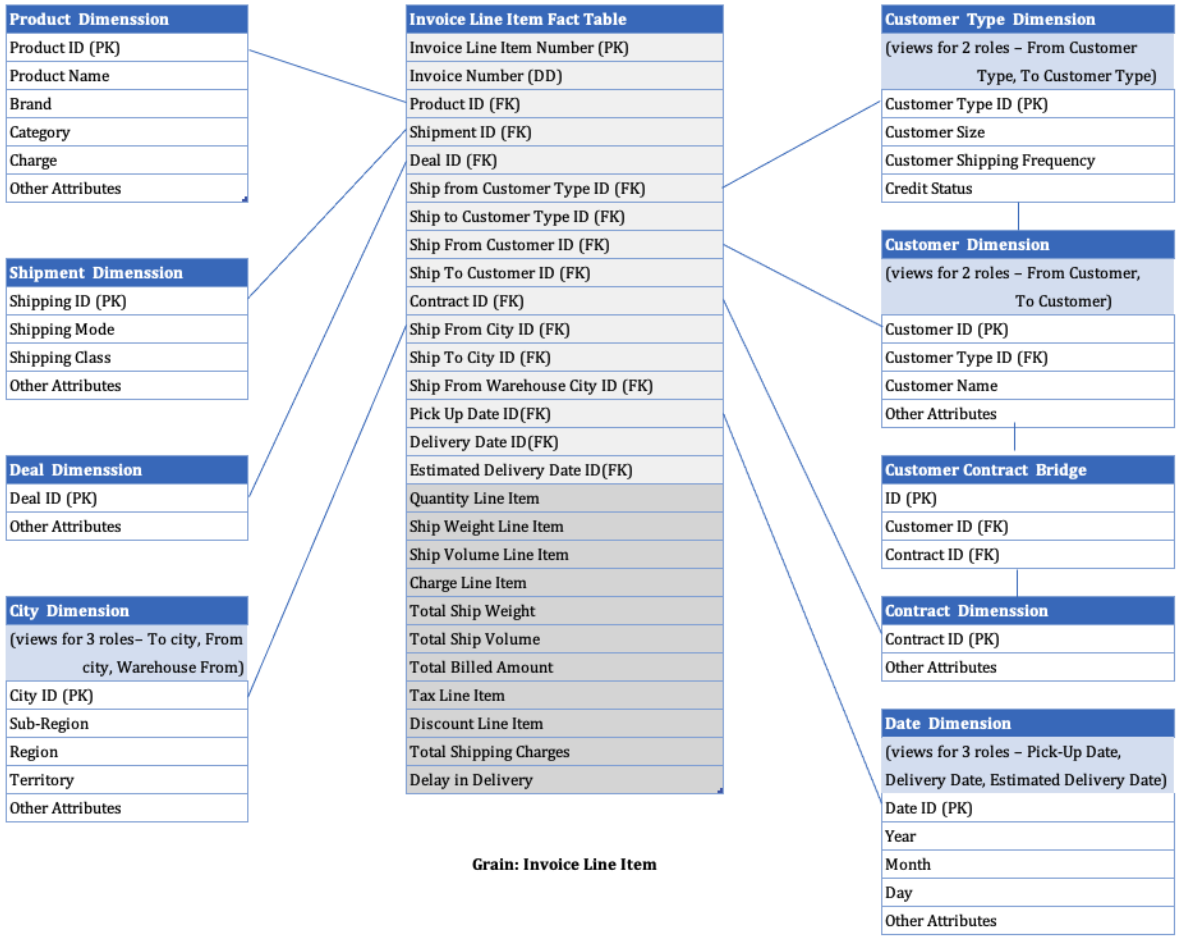
Customer Contract Bridge Dimension
ID (PK)
Customer ID (FK)
Contract ID (FK)

Fact Table:-

The fact table in a shipping dimensional model would contain measures that quantify the shipping activity for a given period of time. Some of the measures that could be included in the fact table are:

- Number of Shipments: This measure would represent the number of shipments that occurred during a particular time period, such as a day, week, or month.
- Total Shipping Cost: This measure would represent the total cost of all shipments during the time period being analysed.
- Total Shipping Revenue: This measure would represent the total revenue generated from shipping during the time period being analysed.
- Average Shipping Cost: This measure would represent the average cost of shipping for all shipments during the time period being analysed.
- Average Shipping Time: This measure would represent the average time it takes for a shipment to reach its destination.

Other measures could be added to the fact table depending on the needs of the business or data warehouse, such as the total weight of the shipments, the number of returns or lost packages, and the average shipping distance. These measures would be related to the dimension table through a foreign key, allowing analysts to aggregate and filter the data to gain insights into their shipping operations.



Grain: Invoice Line Item

Design the Relational Database: -

A star schema can be used to model the relationship between the fact table and dimension tables in a shipping dimensional model. In a star schema, the fact table is located at the centre of the schema, with dimension tables surrounding it. Each dimension table is related to the fact table through a foreign key.

For example, in a shipping dimensional model, the fact table would contain measures such as the number of shipments, total shipping cost, and total shipping revenue. The dimension tables would include attributes such as date, customer, product, shipping method, shipping address, and payment.

The foreign key in the fact table would point to the primary key in each of the dimension tables, allowing analysts to perform queries that aggregate and filter the data along different dimensions. For example, an analyst could use the dimension table for customer to filter the fact table to show only shipping activity for a particular customer. Alternatively, they could use the dimension table for date to analyse shipping activity over a specific time period.

Using the Code: -

- Creating database and tables for Cargo Data Warehouse in MySQL Server.

```
mysql> use cargo;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> show tables;
+-----------------------------------+
| Tables_in_cargo                   |
+-----------------------------------+
| city_dimension                    |
| contract_dimension                |
| customer_contract_bridge_dimension |
| customer_dimension                |
| customer_type_dimension           |
| date_dimension                    |
| deal_dimension                    |
| product_dimension                 |
| shipment_dimension                |
+-----------------------------------+
9 rows in set (0.01 sec)
```

- Inserting and displaying values in the tables above.

```
mysql> use cargo;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> show tables;
+-----------------------------------+
| Tables_in_cargo                   |
+-----------------------------------+
| city_dimension                    |
| contract_dimension                |
| customer_contract_bridge_dimension |
| customer_dimension                |
| customer_type_dimension           |
| date_dimension                    |
| deal_dimension                    |
| product_dimension                 |
| shipment_dimension                |
+-----------------------------------+
9 rows in set (0.01 sec)
```

```
mysql> select * from city_dimension;
+---------+-------------+--------+-----------+
| city_id | sub_region  | region | territory |
+---------+-------------+--------+-----------+
|      30 | Maharashtra | Pune   | India     |
|      31 | M.P         | Bhopal | India     |
+---------+-------------+--------+-----------+
2 rows in set (0.00 sec)

mysql> select * from contract_dimension;
+-------------+
| contract_id |
+-------------+
|          90 |
|          91 |
+-------------+
2 rows in set (0.00 sec)
```

```
mysql> select * from customer_contract_bridge_dimension;
+----+-------------+-------------+
| id | customer_id | contract_id |
+----+-------------+-------------+
| 80 |          60 |          90 |
| 81 |          61 |          91 |
+----+-------------+-------------+
2 rows in set (0.00 sec)

mysql> select * from customer_dimension;
+-------------+------------------+---------------+
| customer_id | customer_type_id | customer_name |
+-------------+------------------+---------------+
|          60 |               20 | Ayush         |
|          61 |               21 | Piyush        |
+-------------+------------------+---------------+
2 rows in set (0.00 sec)
```

```
mysql> select * from customer_type_dimension;
+------------------+---------------+---------------------------+---------------+
| customer_type_id | customer_size | customer_shipping_frquency | credit_status |
+------------------+---------------+---------------------------+---------------+
|               20 |           500 |                      2000 | NIL           |
|               21 |           600 |                      3000 | OVERDUE       |
+------------------+---------------+---------------------------+---------------+
2 rows in set (0.00 sec)

mysql> select * from date_dimension;
+---------+------+-------+---------+
| date_id | year | month | day     |
+---------+------+-------+---------+
|      40 | 2022 | May   | Monday  |
|      41 | 2022 | June  | Tuesday |
+---------+------+-------+---------+
2 rows in set (0.00 sec)
```

```
mysql> select * from deal_dimension;
+---------+
| deal_id |
+---------+
|      70 |
|      71 |
+---------+
2 rows in set (0.00 sec)

mysql> select * from product_dimension;
+------------+--------------+---------+----------+--------+
| product_id | product_name | brand   | category | charge |
+------------+--------------+---------+----------+--------+
|          1 | Air-Jordans  | Nike    | Shoes    |  10000 |
|          2 | Sneakers     | Addidas | Shoes    |   8000 |
+------------+--------------+---------+----------+--------+
2 rows in set (0.01 sec)

mysql> select * from shipment_dimension;
+-------------+---------------+----------------+
| shipping_id | shipping_mode | shipping_class |
+-------------+---------------+----------------+
|          50 | FLIGHT        | First Class    |
|          51 | SHIP          | Second Class   |
+-------------+---------------+----------------+
2 rows in set (0.00 sec)
```

- Creating fact table from the above tables.

```
mysql> select * from fact_invoice_cargo;
+------------+------------+-------------+---------+---------+------------------+-------------+-------------+---------+
| invoice_no | product_id | shipping_id | deal_id | city_id | customer_type_id | customer_id | contract_id | date_id |
+------------+------------+-------------+---------+---------+------------------+-------------+-------------+---------+
|       1000 |          1 |          50 |      70 |      30 |               20 |          60 |          90 |      40 |
|       1001 |          2 |          51 |      71 |      31 |               21 |          61 |          91 |      41 |
+------------+------------+-------------+---------+---------+------------------+-------------+-------------+---------+
2 rows in set (0.00 sec)
```