E-commerce Data Warehouse for Chain of Superstores in Bangladesh

Prepared by

Student ID: 1605023
Name: Ajmain Yasar Ahmed Sahil

SQL Script: https://github.com/FromSaffronCity/ecommerce-data-warehouse/blob/main/src/dw-assignment.sql

Task 1

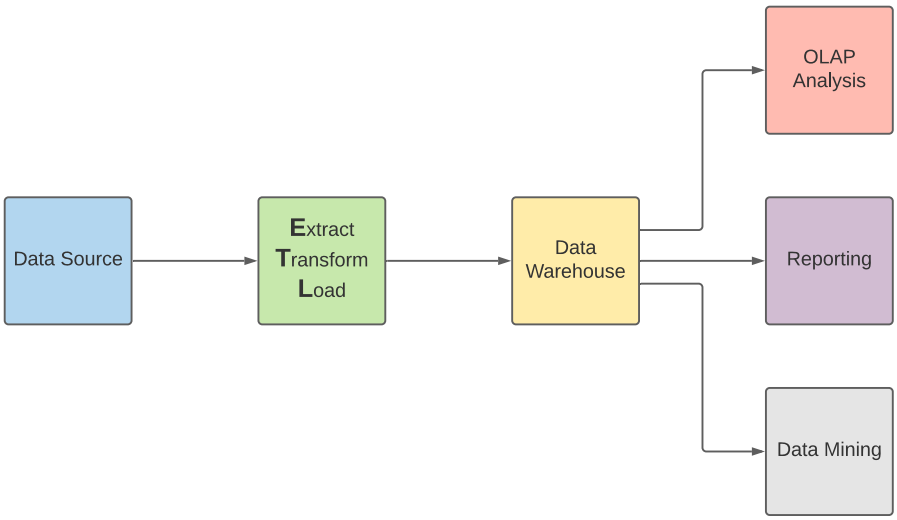
Question

Design the architecture of the data warehouse and explain the sources, preprocessing, noise reduction, transformation, and uploading.

Solution

Data Warehouse is an alternative to data integration in **Big Data Analytics**. The main purpose of data warehouse is to migrate related data from various local data sources to a global/common platform for different data analytics. The following steps are involved in a typical data warehousing process:

1. Carrying out **ETL** process which involves the following:
    1. **Extracting** related data from various local data sources
    2. **Transforming** collected data in local schemas to global schema
    3. **Loading** transformed data into the data warehouse system
2. Carrying out different data analytics on data stored in warehouse



Data Warehouse System Architecture

In data warehousing process, local data sources usually belong to a specific organization. Each of these data sources store similar types of data crucial for the organizational operations. The problem is, data schemas followed and operating **Database Management System (DBMS)** may vary across these local data sources. Consequently, it becomes quite tough to carry out data analytical operations centrally.

Therefore, data from these local data sources are gathered into a global platform. After preprocessing unorganized data and reducing noise from noisy data, the transformation from local schema to global schema is carried out. This conversion is carried out in **Source Driven** manner which means **instead of dropping extra attribute columns from local schema, additional columns are added to the global schema for capturing these extra local attributes**.

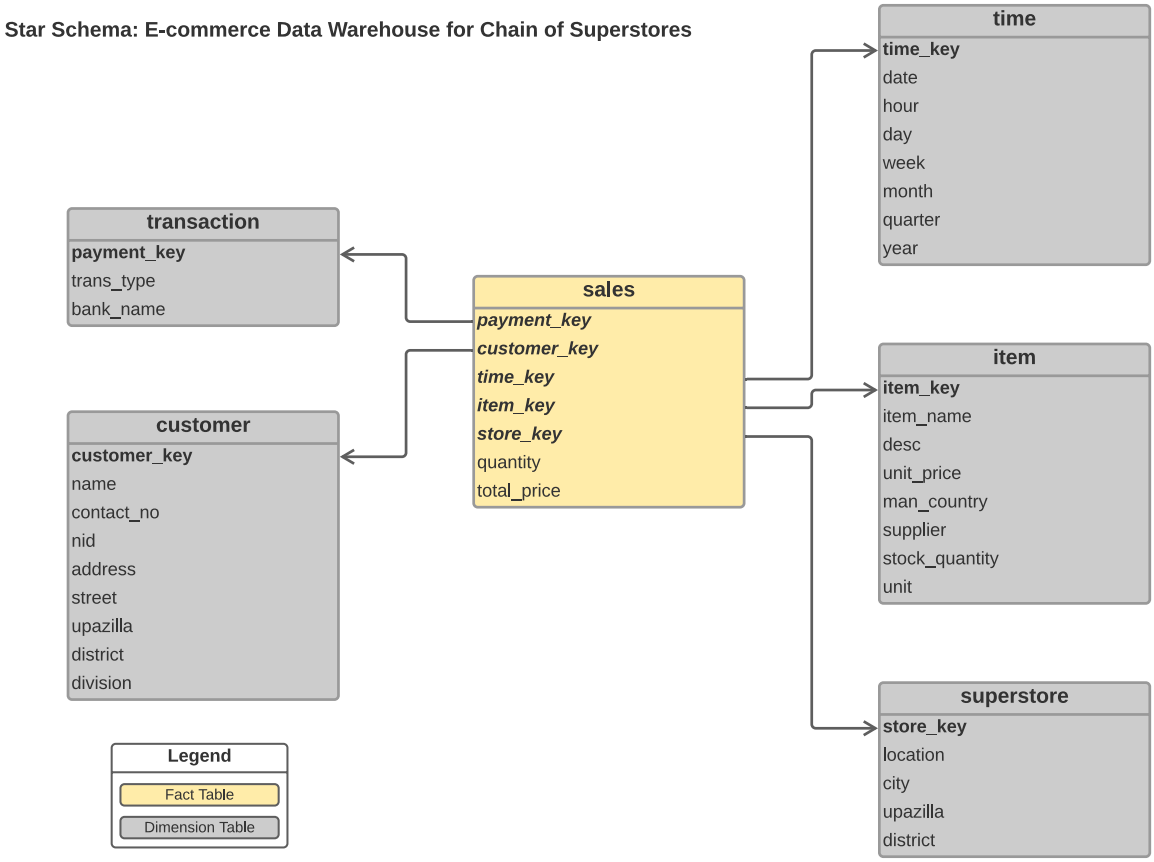
Afterwards, data warehouse system is populated with transformed data.

Task 2

Question

Design the star schema for the data warehouse **using the specified scenario and the datasets provided** and explain how the data of the superstore database will be collected to the data warehouse (source driven or destination driven).

Solution



As mentioned in the solution of **Task 1**, the data from local data sources are collected to the data warehouse in **Source Driven** manner.

Task 3

Question

Implement the star schema using **PostgreSQL** and upload the provided data into the database.

Solution

The star schema is designed and implemented with **PostgreSQL** using the following **SQL** commands:

```
/* creating dimension tables */

/* creating trans_dim table */
create table trans_dim
(payment_key varchar(20) not null primary key,
 trans_type varchar(20),
 bank_name varchar(50));

/* importing trans_dim */
select * from trans_dim;

/* creating customer_dim table */
create table customer_dim
(coustomer_key varchar(20) not null primary key,
 name varchar(50),
 contact_no varchar(20),
 nid varchar(20),
 address varchar(80),
 street varchar(80),
 upazila varchar(20),
 district varchar(20),
 division varchar(20));

/* importing customer_dim */
select * from customer_dim;

/* creating time_dim table */
create table time_dim
(time_key varchar(20) not null primary key,
 date varchar(50),
 hour int,
 day int,
 week varchar(10),
 month integer,
 quarter varchar(10),
 year int);
```

```
/* importing time_dim */
update time_dim
set date = to_timestamp(date, 'DD-MM-YYYY HH24:MI');

alter table time_dim
alter column date type timestamp without time zone
using date::timestamp without time zone;

select * from time_dim;

/* creating item_dim table */
create table item_dim
(item_key varchar(20) not null primary key,
 item_name varchar(50),
 description varchar(50),
 unit_price real,
 man_country varchar(20),
 supplier varchar(50),
 stock_quantity int,
 unit varchar(20));

/* importing item_dim */
select * from item_dim;

/* creating store_dim table */
create table store_dim
(store_key varchar(20) not null primary key,
 location varchar(80),
 city varchar(20),
 upazila varchar(20),
 district varchar(20));

/* importing store_dim */
select * from store_dim;

/* creating fact table */
create table fact_table
(payment_key varchar(20) references trans_dim(payment_key),
 coustomer_key varchar(20) references customer_dim(coustomer_key),
 time_key varchar(20) references time_dim(time_key),
 item_key varchar(20) references item_dim(item_key),
 store_key varchar(20) references store_dim(store_key),
 quantity int,
 unit varchar(20),
 unit_price real,
 total_price real);

/* importing fact_table */
select * from fact_table;
```

The data from provided datasets are uploaded into the corresponding tables using `import data` functionality of **pgAdmin**.

Task 4

Question

Generate three different cross tabulations for three different dimensions using `quantity` / `total_price` attribute. Write **SQL** to find the cross-tabs.

Solution

Cross Tabulation between `trans_type` and `bank_name` from `trans_dim`

```
/* creating cross-tab for transaction dimension */
create table sales_transaction
as
select trans_type, bank_name, total_price
from fact_table, trans_dim
where fact_table.payment_key = trans_dim.payment_key;

select * from sales_transaction;

/* listing corresponding SQL for finding corss-tab */
select sum(total_price) as total_price
from sales_transaction;

select trans_type, sum(total_price) as total_price
from sales_transaction
group by trans_type;

select bank_name, sum(total_price) as total_price
from sales_transaction
group by bank_name;

select trans_type, bank_name, sum(total_price) as total_price
from sales_transaction
group by trans_type, bank_name;
```

Cross Tab

trans_type\bank_name	None	AB Bank Limited	Bangladesh Commerce Bank Limited	...	total_price
cash	2.9210148e+06	0	0	...	2.9210e+06
online	0	2.9437118e+06	2.9443008e+06	...	1.02295e+08
total_price	2.9210148e+06	2.9437118e+06	2.9443008e+06	...	1.05216416e+08

Cross Tabulation between `name` and `division` from `customer_dim`

```
/* creating cross-tab for customer dimension */
create table sales_customer
as
select name, division, total_price
from fact_table, customer_dim
where fact_table.coustomer_key = customer_dim.coustomer_key;

select * from sales_customer;

/* listing corresponding SQL for finding corss-tab */
select sum(total_price) as total_price
from sales_customer;

select name, sum(total_price) as total_price
from sales_customer
group by name;

select division, sum(total_price) as total_price
from sales_customer
group by division;

select name, division, sum(total_price) as total_price
from sales_customer
group by name, division;
```

Cross Tab

name\division	Barishal	Chittagong	Dhaka	Sylhet	total_price
maina devi	0	0	10362.25	0	10362.25
pratibha devi	0	0	12115.75	0	12115.75
mohit maan	0	0	10115.75	0	10115.75
...	...	...	...	...	...
total_price	3.754945e+06	1.1501607e+07	8.627021e+07	3.689525e+06	1.0521683e+08

Cross Tabulation between `item_name` and `man_country` from `item_dim`

```sql
/* creating cross-tab for item dimension */
create table sales_item
as
select item_name, man_country, quantity
from fact_table, item_dim
where fact_table.item_key = item_dim.item_key;

select * from sales_item;

/* listing corresponding SQL for finding corss-tab */
select sum(quantity) as total_quantity
from sales_item;

select item_name, sum(quantity) as total_quantity
from sales_item
group by item_name;

select man_country, sum(quantity) as total_quantity
from sales_item
group by man_country;

select item_name, man_country, sum(quantity) as total_quantity
from sales_item
group by item_name, man_country;
```

Cross Tab

item_name\man_country	Bangladesh	China	India	...	total_quantity
100% Juice Box Variety 6.75 oz	0	0	22939	...	22939
A&W Root Beer - 12 oz cans	0	22132	0	...	22132
A&W Root Beer Diet - 12 oz cans	22183	0	0	...	22183
...	...	...	...	...	...
total_quantity	499902	545114	704943	...	5993859

Task 5

Question

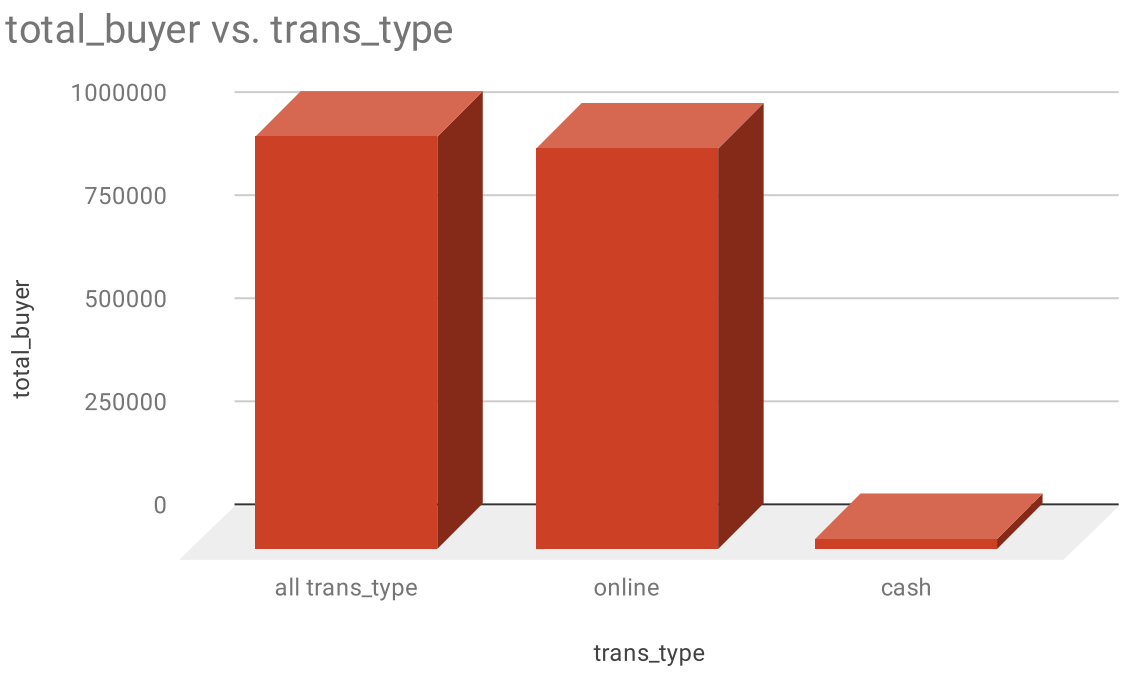
Find and list five important DSS (Decision Support System) reports (one for each dimension) as bar chart. Use `cube` operation in **SQL** to find the DSS report data.

Solution

DSS Report on Transaction Type-wise Buyers Count

```
/* generating DSS report on transaction typewise buyers count */
copy (select coalesce(trans_type, 'all trans_type') trans_type, count(*) as
total_buyer
from fact_table, trans_dim
where fact_table.payment_key = trans_dim.payment_key
group by cube(trans_type)
order by total_buyer desc)
to 'D:\Academic 4-1\CSE453 (High Performance Database System)\dw-assignment\dw-
assignment-report\csv\dss_trans_type_buyer.csv'
delimiter ',' csv header;
```

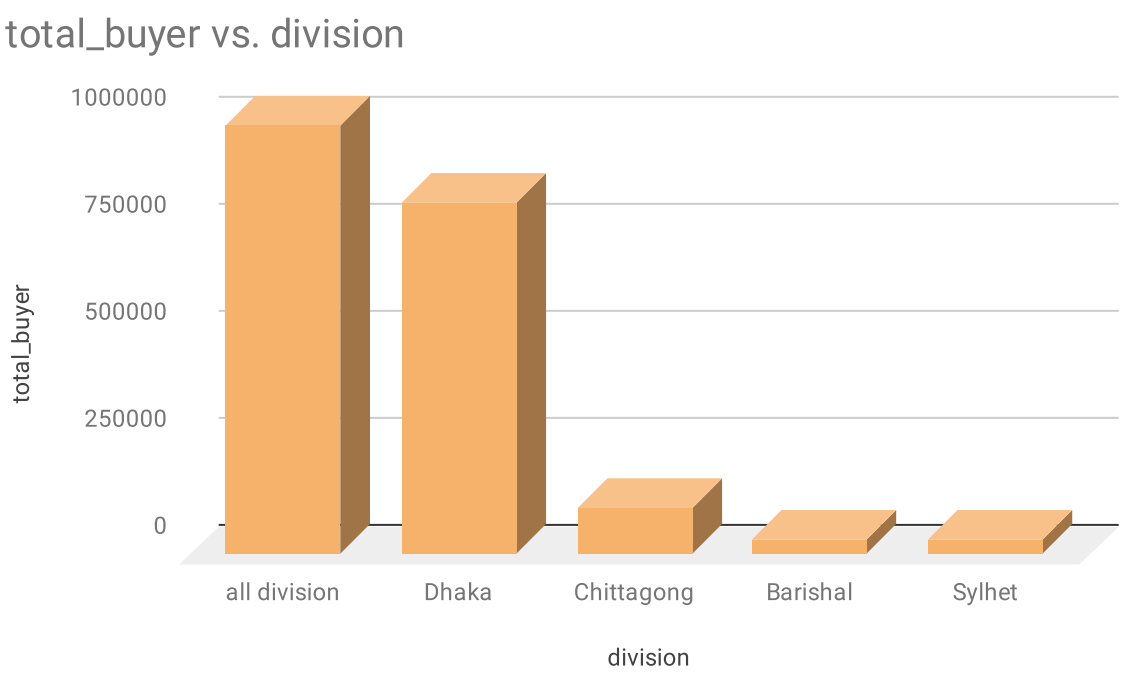
Bar Chart



DSS Report on Division-wise Buyers Count

```
/* generating DSS report on divisionwise buyers count */
copy (select coalesce(division, 'all division') division, count(*) as
total_buyer
from fact_table, customer_dim
where fact_table.coustomer_key = customer_dim.coustomer_key
group by cube(division)
order by total_buyer desc)
to 'D:\Academic 4-1\CSE453 (High Performance Database System)\dw-assignment\dw-
assignment-report\csv\dss_division_buyer.csv'
delimiter ',' csv header;
```

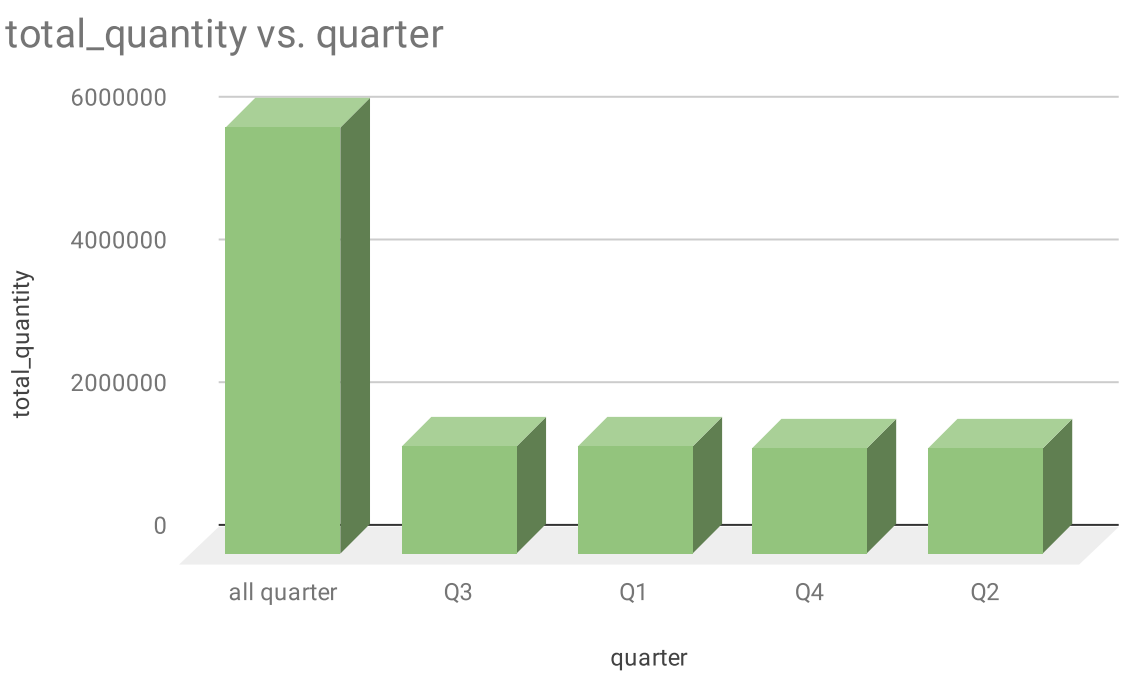
Bar Chart



DSS Report on Quarter-wise Sales Count

```
/* generating DSS report on quarterwise sales count */
copy (select coalesce(quarter, 'all quarter') quarter, sum(quantity) as
total_quantity
from fact_table, time_dim
where fact_table.time_key = time_dim.time_key
group by cube(quarter)
order by total_quantity desc)
to 'D:\Academic 4-1\CSE453 (High Performance Database System)\dw-assignment\dw-
assignment-report\csv\dss_quarter_count.csv'
delimiter ',' csv header;
```

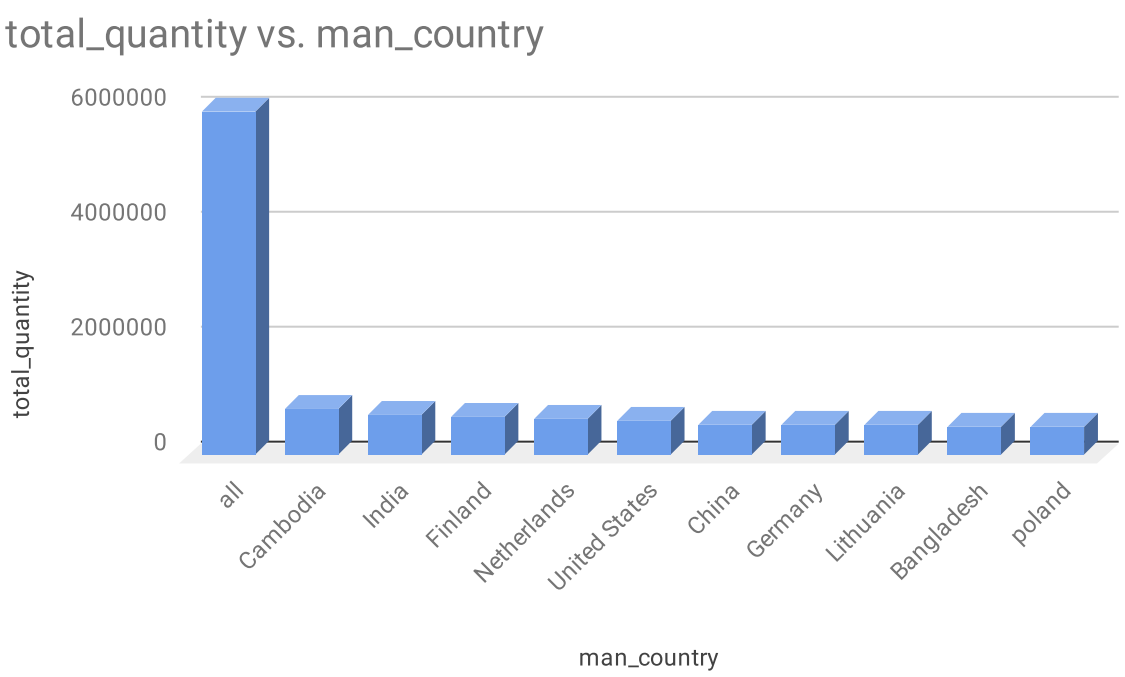
Bar Chart



DSS Report on Manufacturer Country-wise Sales Count

```
/* generating DSS report on manufacturer countrywise sales count */
copy (select coalesce(man_country, 'all man_country') man_country, sum(quantity)
as total_quantity
from fact_table, item_dim
where fact_table.item_key = item_dim.item_key
group by cube(man_country)
order by total_quantity desc)
to 'D:\Academic 4-1\CSE453 (High Performance Database System)\dw-assignment\dw-
assignment-report\csv\dss_man_country_count.csv'
delimiter ',' csv header;
```

bar Chart



DSS Report on District-wise Sales Earning

```
/* generating DSS report on districtwise sales earning */
copy (select coalesce(district, 'all district') district, sum(total_price) as
total_price
from fact_table, store_dim
where fact_table.store_key = store_dim.store_key
group by cube(district)
order by total_price desc)
to 'D:\Academic 4-1\CSE453 (High Performance Database System)\dw-assignment\dw-
assignment-report\csv\dss_district_earning.csv'
delimiter ',' csv header;
```

Bar Chart