PES University

Department of Computer Science

UE17ECS312 - Database Technologies

Assigment -1 Report

Shreyas S Mavanoor
SEM V 'E'
PES1201700837

**Assignment: Build a star schema data warehouse for a sample database**

Introduction

The transaction system chosen for building a star schema data warehouse is the DVD RENTAL database for learning Postgresql database available at:
http://www.postgresqltutorial.com/wp-content/uploads/2019/05/dvdrental.zip

The star schema data model

Put the dimension and fact table schema here.
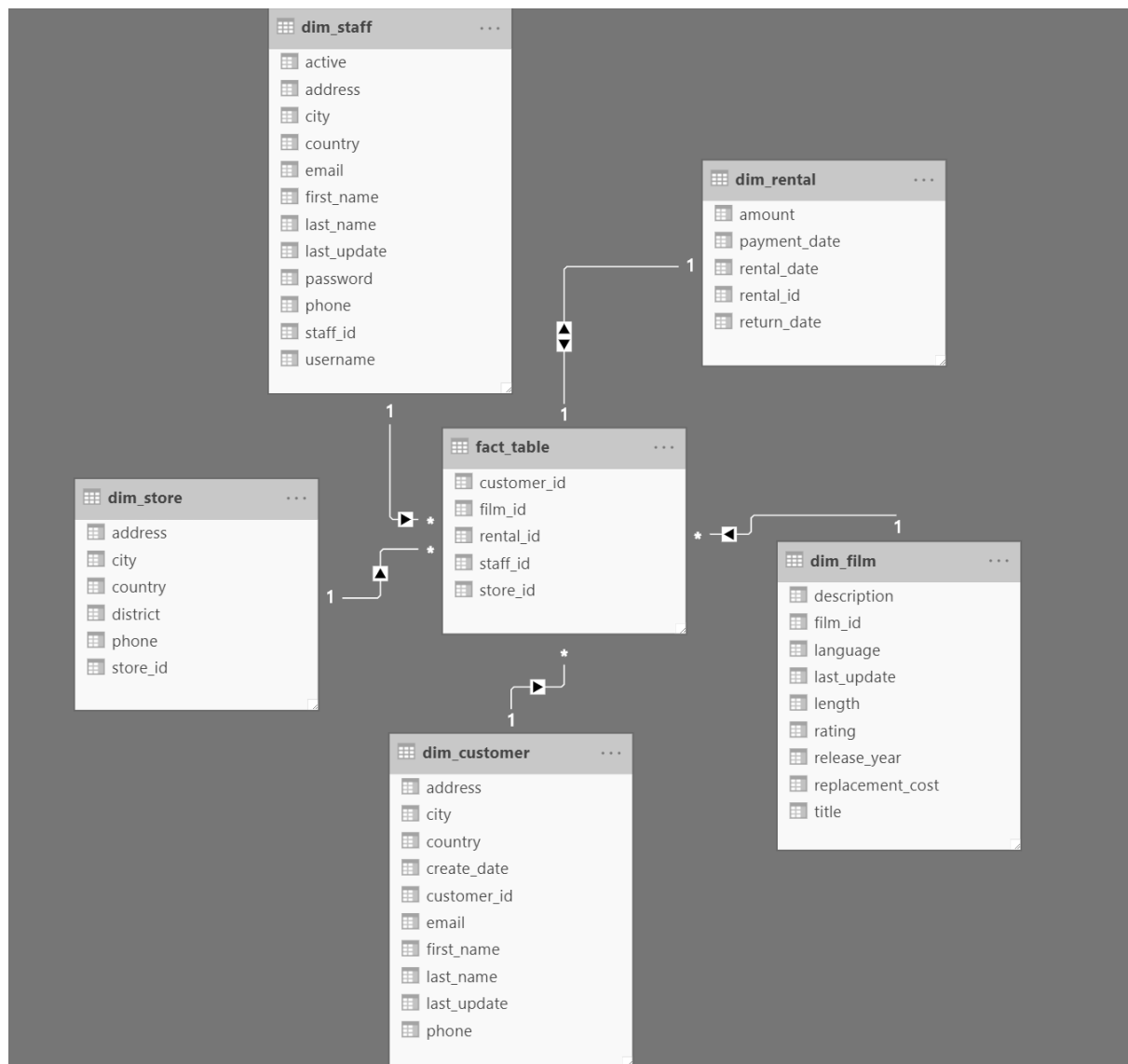
Time Dimension Table

```
create table dim_time(

time_id SERIAL primary key,

c_date date not null,

c_year integer,

c_month integer,

c_week integer,

c_qtr integer,

c_day_of_week integer,

c_day_of_year integer);
```

```sql
CREATE TABLE DIM_CUSTOMER
(
CUSTOMER_ID INT,
FIRST_NAME VARCHAR(50),
LAST_NAME VARCHAR(50),
EMAIL VARCHAR(50),
PHONE VARCHAR(20),
ADDRESS VARCHAR(50),
CITY VARCHAR(50),
COUNTRY VARCHAR(50),
CREATE_DATE date NOT NULL,
last_update timestamp,
PRIMARY KEY (CUSTOMER_ID)
);

CREATE TABLE DIM_STORE(
STORE_ID INT,
ADDRESS      VARCHAR(60),
CITY VARCHAR(50),
DISTRICT VARCHAR(60),
COUNTRY VARCHAR(50),
PHONE VARCHAR(20),
PRIMARY KEY(STORE_ID)
);



CREATE TABLE DIM_STAFF
(
STAFF_ID INT NOT NULL,
FIRST_NAME VARCHAR(45)NOT NULL,
LAST_NAME VARCHAR(45)NOT NULL,
```

```sql
ADDRESS VARCHAR(60) NOT NULL,

CITY VARCHAR(50),

COUNTRY VARCHAR(50),

EMAIL VARCHAR(50),

PHONE VARCHAR(20),

ACTIVE BOOLEAN NOT NULL DEFAULT true,

USERNAME VARCHAR(16) NOT NULL,

PASSWORD VARCHAR(40),

last_update timestamp NOT NULL DEFAULT now(),

PRIMARY KEY (STAFF_ID)

);



CREATE TABLE DIM_RENTAL

(

RENTAL_ID INT NOT NULL,

RENTAL_DATE timestamp NOT NULL,

RETURN_DATE timestamp,

AMOUNT INT,

PAYMENT_DATE timestamp,

PRIMARY KEY (RENTAL_ID)

);



CREATE TABLE FACT_TABLE

(

CUSTOMER_ID INT,

FILM_ID INT,

RENTAL_ID INT,

STORE_ID INT,

STAFF_ID INT NOT NULL,

FOREIGN KEY (CUSTOMER_ID) REFERENCES DIM_CUSTOMER (CUSTOMER_ID),

FOREIGN KEY (FILM_ID) REFERENCES DIM_FILM (FILM_ID),
```

FOREIGN KEY (RENTAL_ID) REFERENCES DIM_RENTAL (RENTAL_ID),

FOREIGN KEY (STORE_ID) REFERENCES DIM_STORE (STORE_ID),

FOREIGN KEY (STAFF_ID) REFERENCES DIM_STAFF (STAFF_ID)

);

The ETL scripts

Put the ETL sql statements/scripts here:

```
--CUSTOMER
INSERT INTO DIM_CUSTOMER
(
SELECT
CU.CUSTOMER_ID,CU.FIRST_NAME,CU.LAST_NAME,CU.EMAIL,AD.PHONE,AD.ADDR
ESS,CITY.CITY,COU.COUNTRY,CU.CREATE_DATE,CU.last_update
FROM CUSTOMER CU JOIN  ADDRESS AD ON (CU.ADDRESS_ID =AD.ADDRESS_ID)
JOIN CITY ON (CITY.CITY_ID=AD.CITY_ID)
JOIN COUNTRY COU ON (COU.COUNTRY_ID=CITY.COUNTRY_ID)
);

--TIME
insert into dim_time(c_date)
 select generate_series(DATE '20060101', DATE '20091231', interval '1' day) as cal_date;

update dim_time
  set c_year = EXTRACT(YEAR from c_date),
        c_month = EXTRACT(MONTH from c_date),
        c_week = EXTRACT(WEEK from c_date),
        c_qtr = EXTRACT(QUARTER from c_date),
        c_day_of_week = EXTRACT(DOW from c_date),
        c_day_of_year = EXTRACT(DOY from c_date);


--FILM
INSERT INTO DIM_FILM
(
SELECT
F.FILM_ID,F.TITLE,F.DESCRIPTION,F.RELEASE_YEAR,L.NAME,F.last_update,F.LENGT
H,F.REPLACEMENT_COST,F.RATING
FROM FILM F JOIN LANGUAGE L ON (L.LANGUAGE_ID=F.LANGUAGE_ID)
);

--
```

```sql
INSERT INTO DIM_STORE
(
SELECT
S.STORE_ID,AD.ADDRESS,CITY.CITY,AD.DISTRICT,COU.COUNTRY,AD.PHONE
FROM STORE S JOIN  ADDRESS AD ON (S.ADDRESS_ID = AD.ADDRESS_ID)
JOIN CITY ON (CITY.CITY_ID=AD.CITY_ID)
JOIN COUNTRY COU ON (COU.COUNTRY_ID=CITY.COUNTRY_ID)
);

INSERT INTO DIM_STAFF
(
SELECT
ST.STAFF_ID,ST.FIRST_NAME,ST.LAST_NAME,AD.ADDRESS,CITY.CITY,COU.COUNT
RY,ST.EMAIL,AD.PHONE,ST.ACTIVE,ST.USERNAME,ST.PASSWORD,ST.last_update
FROM STAFF ST JOIN  ADDRESS AD ON (ST.ADDRESS_ID = AD.ADDRESS_ID)
JOIN CITY ON (CITY.CITY_ID=AD.CITY_ID)
JOIN COUNTRY COU ON (COU.COUNTRY_ID=CITY.COUNTRY_ID)
);

--RENTAL
INSERT INTO DIM_RENTAL
(
SELECT R.RENTAL_ID,R.RENTAL_DATE,R.return_date,P.AMOUNT,P.PAYMENT_DATE
FROM RENTAL R JOIN PAYMENT P ON (P.RENTAL_ID=R.RENTAL_ID)
WHERE R.RENTAL_ID != 4591
);




--FACT TABLE
INSERT INTO FACT_TABLE
(
SELECT R.CUSTOMER_ID,F.FILM_ID,R.RENTAL_ID,S.STORE_ID,P.STAFF_ID
FROM RENTAL R JOIN PAYMENT P ON P.RENTAL_ID=R.RENTAL_ID
JOIN INVENTORY I ON (R.INVENTORY_ID=I.INVENTORY_ID)
JOIN FILM F ON (I.FILM_ID=F.FILM_ID)
JOIN STAFF ST ON (ST.STAFF_ID=P.STAFF_ID)
JOIN STORE S ON (ST.STORE_ID=S.STORE_ID)
WHERE R.RENTAL_ID != 4591
);
```

Time Dimension Table

Populate the time dimension table

```
insert into dim_time(c_date)
 select generate_series(DATE '20050101', DATE '20061231', interval '1' day) as cal_date;

update dim_time
  set c_year = EXTRACT(YEAR from c_date),
        c_month = EXTRACT(MONTH from c_date),
        c_week = EXTRACT(WEEK from c_date),
        c_qtr = EXTRACT(QUARTER from c_date),
        c_day_of_week = EXTRACT(DOW from c_date),
        c_day_of_year = EXTRACT(DOY from c_date);
```

Tests

Check the record counts in source and target tables. Also, it is a good idea to test if the queries return the same values. Eg. rentals for a month should be the same in both source(OLTP) and target (DW)

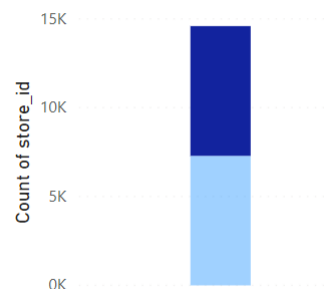| Dimension Table	No of Rows	Source Table	No of Rows
dim_film	1000	1000	1000
dim_customer	599	599	599
dim_staff	2	2	2

Reports and Dashboard

Paste the screen shots of Microsoft Power BI Reports and dashboards.
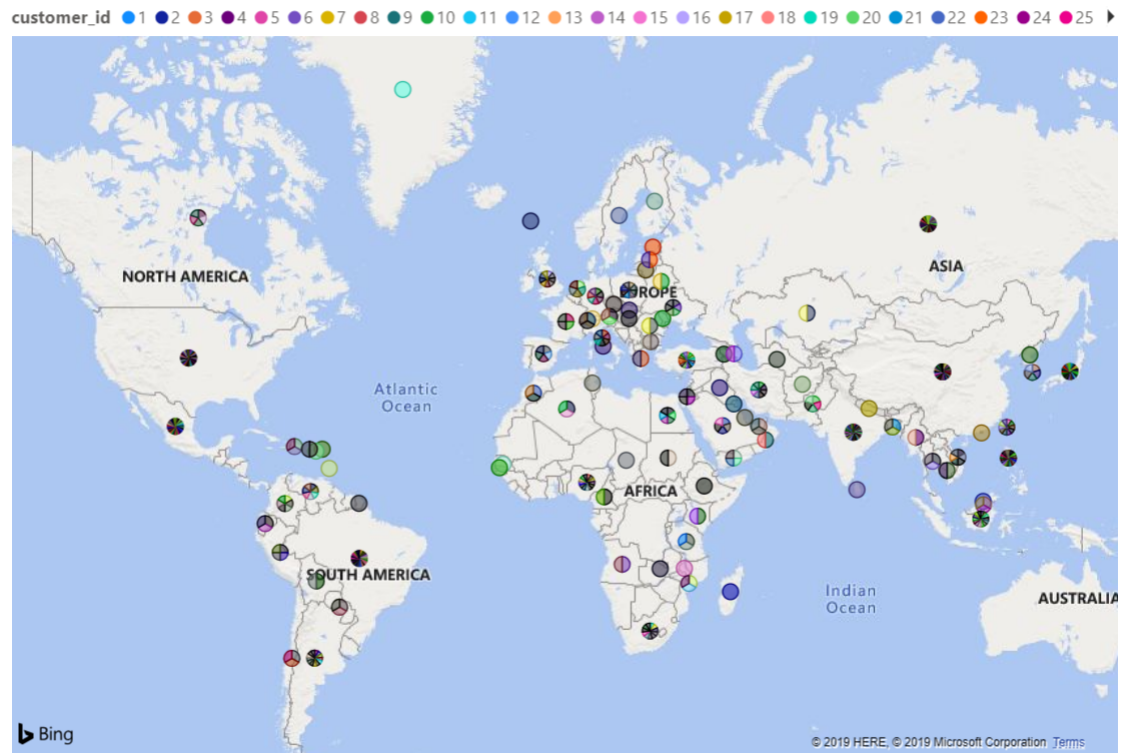Briefly describe the graphs and charts.

This chart shows the no. of stores by city.

country and customer_id

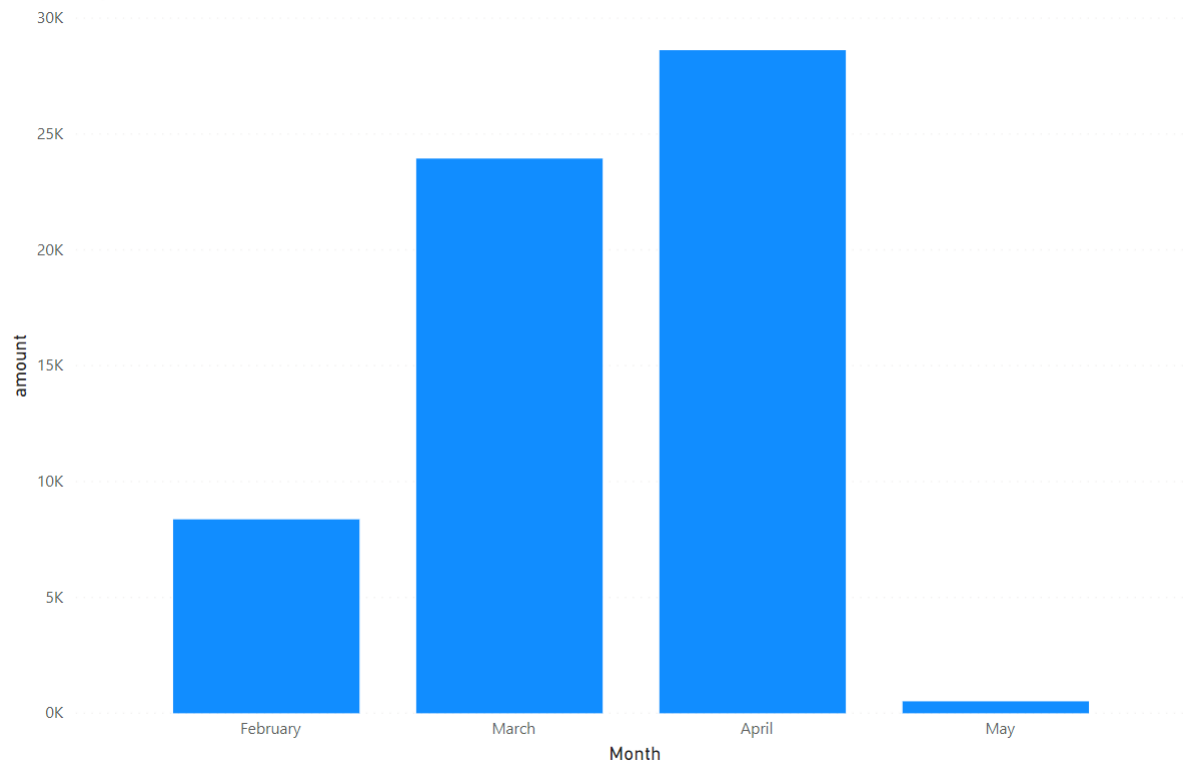This chart shows the spread of customers across the globe.

city and store_id

store_id ●1 ●2

This graph shows the stores and their respective cities.

amount by Month

This histogram gives the amount of rentals by month.

Conclusion

Power BI provides immense capability to showcase the data facts through great visualizations.